ORACLE

Cloud Infrastructure

User Guide

5/24/2019

Copyright © 2016, 2019, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement

between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

CONTENTS

CHAPTER 1 About Oracle Cloud Infrastructure 16
Prefer Online Help? 18
Need API Documentation? 18

CHAPTER 2 Service Essentials 19
Security Credentials 20
Regions and Availability Domains 22
IP Address Ranges 29
Resource Identifiers 35
Resource Tags 39
Resource Monitoring 44
Service Limits 60
Console Announcements 78
Prerequisites for Oracle Platform Services on Oracle Cloud Infrastructure 86
Billing and Payment Tools Overview 99
Console Cookies and Local Storage 112
My Services Use Cases 115

CHAPTER 3 Archive Storage 168
Overview of Archive Storage 168

CHAPTER 4 Audit 174
Overview of Audit 174
Contents of an Audit Log Event 175

Viewing Audit Log Events ... 180
Setting Audit Log Retention Period ... 185
Bulk Export of Audit Log Events ... 186

CHAPTER 5 Block Volume ... 191

Overview of Block Volume ... 191
iSCSI Commands and Information ... 199
Volume Groups ... 200
Creating a Volume ... 212
Attaching a Volume ... 214
Connecting to a Volume ... 220
Listing Volumes ... 229
Listing Volume Attachments ... 231
Listing Boot Volume Attachments ... 232
Renaming a Volume ... 233
Resizing a Volume ... 234
Overview of Block Volume Backups ... 243
Backing Up a Volume ... 252
Policy-Based Backups ... 254
Copying a Volume Backup Between Regions ... 258
Restoring a Backup to a New Volume ... 262
Cloning a Volume ... 264
Disconnecting From a Volume ... 268
Detaching a Volume ... 269
Deleting a Volume ... 271
Block Volume Performance ... 272
Block Volume Metrics ... 291

CHAPTER 6 Compute ... 295

Overview of the Compute Service ... 295
Best Practices for Your Compute Instance ... 300
Protecting Data on NVMe Devices ... 312

Boot Volumes ... 324
Oracle-Provided Images ... 362
Compute Shapes ... 378
Installing and Running Oracle Ksplice ... 384
Managing Custom Images ... 386
Image Import/Export ... 396
Bring Your Own Image (BYOI) ... 403
OS Kernel Updates ... 424
Managing Key Pairs on Linux Instances ... 427
Creating an Instance ... 430
Managing Compute Instances ... 444
Autoscaling ... 459
Connecting to an Instance ... 468
Instance Console Connections ... 473
Adding Users on an Instance ... 482
Displaying the Console for an Instance ... 485
Getting Instance Metadata ... 486
Updating Instance Metadata ... 488
Renaming an Instance ... 491
Moving a Compute Instance ... 491
Migrating an Instance from a Local to Remote Boot Volume ... 498
Stopping and Starting an Instance ... 502
Terminating an Instance ... 506
Enabling Monitoring for Compute Instances ... 508
Compute Metrics ... 522
Compute Performance ... 533
Compute Health Monitoring for Bare Metal Instances ... 536

CHAPTER 7 Container Engine for Kubernetes ... 540
Overview of Container Engine for Kubernetes ... 540
Preparing for Container Engine for Kubernetes ... 542
About Kubernetes Clusters and Nodes ... 577

Creating a Kubernetes Cluster ... 579
Downloading a kubeconfig File to Enable Cluster Access ... 588
Modifying a Kubernetes Cluster ... 591
Deleting a Kubernetes Cluster ... 593
Monitoring Clusters ... 594
Monitoring Operations of Container Engine for Kubernetes ... 597
Accessing a Cluster Using kubectl ... 598
Starting the Kubernetes Dashboard ... 598
Deploying a Sample Nginx App on a Cluster Using kubectl ... 599
Pulling Images from Registry during Deployment ... 601
Connecting to Worker Nodes Using SSH ... 603
About Access Control and Container Engine for Kubernetes ... 606
Kubernetes Versions and Container Engine for Kubernetes ... 611
Creating Load Balancers to Distribute Traffic Between Cluster Nodes ... 618
Creating a Persistent Volume Claim ... 624
Adding OCI Service Broker for Kubernetes to Clusters ... 628
Example: Setting Up an Ingress Controller on a Cluster ... 630

CHAPTER 8 Data Transfer ... 640
Overview of Data Transfer Service ... 640
Data Transfer Disk ... 641
Data Transfer Appliance ... 647
Preparing for Data Transfer ... 654
Installing and Configuring the Data Transfer Utility ... 657
Managing Disk Data Transfers ... 665
Managing Appliance Data Transfers ... 692

CHAPTER 9 Database ... 734
Overview of the Database Service ... 734
Exadata DB Systems ... 737
Bare Metal and Virtual Machine DB Systems ... 845
Overview of Autonomous Database ... 1123

Migrating Databases to the Cloud 1153
Troubleshooting 1216

CHAPTER 10 Edge Services 1245
Overview of the DNS Service 1245
Overview of the Health Checks Service 1275
Overview of the Traffic Management Steering Policies Service 1291
Overview of the Web Application Firewall Service 1341

CHAPTER 11 Email Delivery 1419
Overview of the Email Delivery Service 1419
Generate SMTP Credentials for a User 1426
Managing Approved Senders 1427
Configure SPF 1429
Configure SMTP Connection 1430
Managing the Suppression List 1432
Deliverability Best Practices 1434

CHAPTER 12 File Storage 1440
Overview of File Storage 1440
About Security 1447
Creating File Systems 1469
Mounting File Systems 1482
Managing File Systems 1513
Managing Mount Targets 1528
Managing Snapshots 1565
Paths in File Systems 1571
Troubleshooting Your File System 1573

CHAPTER 13 IAM 1593
Overview of Oracle Cloud Infrastructure Identity and Access Management 1593
Getting Started with Policies 1609
How Policies Work 1612

Common Policies ... 1622
Advanced Policy Features ... 1642
Policy Syntax ... 1647
Policy Reference ... 1653
User Credentials ... 1831
Federating with Identity Providers ... 1834
User Provisioning for Federated Users ... 1892
Managing User Capabilities for Federated Users ... 1898
Calling Services from an Instance ... 1902
Managing Users ... 1909
Managing Groups ... 1919
Managing Dynamic Groups ... 1924
Managing Compartments ... 1935
Managing Tags and Tag Namespaces ... 1944
Managing Regions ... 1973
Managing the Tenancy ... 1977
Managing Policies ... 1980
Managing User Credentials ... 1987
Managing Authentication Settings ... 2000
Managing Multi-Factor Authentication ... 2004

CHAPTER 14 Key Management ... 2014
Overview of Key Management ... 2014
Managing Keys ... 2018
Managing Vaults ... 2040
Using Keys ... 2048

CHAPTER 15 Load Balancing ... 2052
Overview of Load Balancing ... 2052
How Load Balancing Policies Work ... 2066
Connection Management ... 2068
HTTP "X-" Headers ... 2071

Session Persistence 2072
Managing a Load Balancer 2075
Managing Backend Sets 2092
Managing Backend Servers 2099
Managing Load Balancer Listeners 2107
Managing Request Routing 2113
Managing Rule Sets 2127
Managing SSL Certificates 2134
Editing Health Check Policies 2145
Viewing the State of a Work Request 2152
Load Balancing Metrics 2154

CHAPTER 16 Marketplace 2169
Overview of Marketplace 2169
Working with Listings 2170

CHAPTER 17 Monitoring 2174
Monitoring Overview 2174
Viewing Default Metric Charts 2191
Building Metric Queries 2225
Publishing Custom Metrics 2250
Managing Alarms 2254
Best Practices for your Alarms 2279
Monitoring Query Language (MQL) Reference 2282

CHAPTER 18 Networking 2291
Overview of Networking 2291
Scenario A: Public Subnet 2304
Scenario B: Private Subnet with a VPN 2313
Scenario C: Public and Private Subnets with a VPN 2329
Advanced Scenario: Transit Routing 2349
VCNs and Subnets 2372
Ways to Secure Your Network 2384

Access Control 2386
Security Lists 2389
Virtual Network Interface Cards (VNICs) 2406
Private IP Addresses 2422
Public IP Addresses 2440
DNS in Your Virtual Cloud Network 2454
DHCP Options 2464
Route Tables 2473
Dynamic Routing Gateways (DRGs) 2484
VPN Connect 2492
VPN Connect Overview 2493
Supported IPSec Parameters 2505
Setting Up VPN Connect 2506
Configuring Your CPE 2541
Verified CPE Devices 2665
Working with VPN Connect 2666
VPN Connect FAQ 2676
Using the API for VPN Connect 2677
VPN Connect Troubleshooting 2679
FastConnect 2690
FastConnect Overview 2690
FastConnect Requirements 2706
FastConnect: With an Oracle Provider 2712
FastConnect: With a Third-Party Provider 2733
FastConnect: Colocation with Oracle 2759
FastConnect Troubleshooting 2783
Internet Gateway 2786
NAT Gateway 2796
Access to Oracle Services: Service Gateway 2805
Access to Other VCNs: Peering 2822
Local VCN Peering (Within Region) 2826
Remote VCN Peering (Across Regions) 2849

Access to Oracle Cloud Infrastructure Classic ... 2865
Connection Over Oracle Network ... 2866
Connection Over IPSec VPN ... 2875
Access to Other Clouds with Libreswan ... 2883
Network Performance ... 2899
Networking Metrics ... 2901
Troubleshooting ... 2909

CHAPTER 19 Notifications ... 2921
Notifications Overview ... 2921
Managing Topics and Subscriptions ... 2925
Publishing Messages ... 2932
Notifications Metrics ... 2934

CHAPTER 20 Object Storage ... 2939
Overview of Object Storage ... 2939
Understanding Object Storage Namespaces ... 2945
Managing Buckets ... 2947
Managing Objects ... 2968
Copying Objects ... 2988
Using Pre-Authenticated Requests ... 2996
Using Multipart Uploads ... 3005
Using Object Lifecycle Management ... 3011
Object Storage Metrics ... 3022
Hadoop Support ... 3026
Designating Compartments for the Amazon S3 Compatibility and Swift APIs ... 3027
Amazon S3 Compatibility API ... 3031

CHAPTER 21 Registry ... 3035
Overview of Registry ... 3035
Preparing for Registry ... 3037
About Images ... 3039
About Repositories ... 3039

Creating a Repository ... 3040
Pushing Images Using the Docker CLI ... 3041
Pulling Images Using the Docker CLI ... 3046
Pulling Images from Registry during Kubernetes Deployment ... 3048
Viewing Images and Image Details ... 3050
Deleting an Image ... 3052
Retaining and Deleting Images Using Retention Policies ... 3052
Deleting a Repository ... 3059
Getting an Auth Token ... 3060
Policies to Control Repository Access ... 3060

CHAPTER 22 Overview of Resource Manager ... 3064
Authentication and Authorization ... 3065
Tagging Resources ... 3065
Key Concepts ... 3066
Generalized Workflow ... 3069
Using Resource Manager in the Console ... 3071
Using Resource Manager with the CLI ... 3082
Using Remote Exec ... 3092
Sample: Creating a Compute Instance Using Resource Manager ... 3096

CHAPTER 23 Search ... 3112
Overview of Search ... 3112
Search Language Syntax ... 3115
Sample Queries ... 3124
Querying Resources ... 3128
Troubleshooting Search ... 3134

CHAPTER 24 Security Guide and Announcements ... 3136
Oracle Cloud Infrastructure Security Guide ... 3136
Oracle Cloud Security Response to Intel L1TF Vulnerabilities ... 3249
Oracle Cloud Security Response to Intel Microarchitectural Data Sampling (MDS) Vulnerabilities ... 3258

CHAPTER 25 Storage Gateway 3270
Overview of Storage Gateway 3270
Features of Storage Gateway 3278
Getting Started With Storage Gateway 3281
Configuring the Cache for File Systems 3282
Understanding Storage Gateway Performance 3290
Interacting With Object Storage 3293
Installing Storage Gateway 3297
Logging In to the Storage Gateway Management Console 3310
Creating Your First File System 3312
Managing File Systems 3318
Managing Storage Gateway 3330
Using Storage Gateway File Management Operations 3335
Monitoring Storage Gateway 3342
Using Storage Gateway Cloud Sync 3346
Best Practices for Using Storage Gateway 3354
Troubleshooting Storage Gateway 3354
Upgrading Storage Gateway 3358
Uninstalling Storage Gateway 3365
Getting Help with Storage Gateway 3367
CHAPTER 26 Streaming Service Overview 3369
Streaming Usage Scenarios 3369
Streaming Concepts 3369
How Streaming Works 3371
Limits on Streaming Resources 3372
Resource Identifiers 3372
Ways to Access Oracle Cloud Infrastructure 3373
Using Streaming 3373
Authentication and Authorization 3373
Tagging Resources 3374
Managing Streams 3374

Publishing Messages ... 3377
Consuming Messages ... 3378
Using the Streaming SDK ... 3381
Streaming Metrics ... 3390

CHAPTER 27 Developer Tools ... 3399
Software Development Kits and Command Line Interface ... 3399
Other Tools and Plug-ins ... 3469
Tools Configuration ... 3550
REST APIs ... 3560

GLOSSARY ... 3651

RELEASE NOTES ... 3672

CHAPTER 1 About Oracle Cloud Infrastructure

Oracle Cloud Infrastructure provides bare metal cloud infrastructure that lets you create networking, compute, and storage resources for your enterprise workloads.

If you're new to Oracle Cloud Infrastructure and would like to learn some key concepts and take a quick tutorial, see the *Oracle Cloud Infrastructure Getting Started Guide*.

If you're ready to create cloud resources such as users, access controls, cloud networks, instances, and storage volumes, this guide is right for you. It provides the following information about using Oracle Cloud Infrastructure:

Service	What's Covered	Chapter
Archive Storage	Preserving cold data.	Archive Storage
Audit	Logging activity in your cloud.	Audit
Block Volume	Adding storage capacity to instances.	Block Volume
Compute	Launching compute instances and connecting to them by using an SSH key pair.	Compute
Container Engine for Kubernetes	Defining and creating Kubernetes clusters to enable the deployment, scaling, and management of containerized applications.	Container Engine for Kubernetes
Data Transfer	Migrating large volumes of data.	Data Transfer
Database	Launching DB Systems and managing Oracle databases.	Database

Service	What's Covered	Chapter
Edge Services	Encompasses several services that allow you to manage, secure, and maintain your domains and endpoints.	Edge Services
Email Delivery	Sending large volume email.	Email Delivery
File Storage	Managing shared file systems, mount targets, and snapshots.	File Storage
IAM	Setting up administrators, users, and groups and specifying their permissions to access to cloud resources.	Key Management
Key Management	Creating and managing encryption keys and key vaults to control the encryption of your data.	Key Management
Load Balancing	Setting up load balancers, listeners, backend sets, certificate bundles, and managing health check policies.	Load Balancing
Monitoring	Querying metrics and managing alarms to monitor the health, capacity, and performance of your cloud resources.	Monitoring
Networking	Setting up cloud networks, subnets, gateways, route tables, and security lists.	Networking
Notifications	Setting up topics and subscriptions, and publishing messages.	Notifications

Service	What's Covered	Chapter
Object Storage	Creating and managing buckets to store objects, and uploading and accessing data files.	Object Storage
Registry	Storing, sharing, and managing development artifacts like Docker images in an Oracle-managed registry.	Registry
Search	Searching for Oracle Cloud Infrastructure resources using free text search or advanced queries.	Search

For a description of the terminology used throughout this guide, see the GLOSSARY.

Prefer Online Help?

The information in this guide and the *Getting Started Guide* is also available in the online help at https://docs.cloud.oracle.com/iaas/Content/home.htm.

Need API Documentation?

For general information, see REST APIs. For links to the detailed service API documentation, see the online help at https://docs.cloud.oracle.com/iaas/Content/home.htm.

CHAPTER 2 Service Essentials

The following topics provide essential information that applies across Oracle Cloud Infrastructure.

Security Credentials

The types of credentials you'll use when working with Oracle Cloud Infrastructure.

Regions and Availability Domains

An introduction to the concepts of regions and availability domains.

Resource Identifiers

A description of the different ways your Oracle Cloud Infrastructure resources are identified.

Resource Tags

Information about Oracle Cloud Infrastructure tags and how to apply them to your resources.

Service Limits

A list of the default limits applied to your cloud resources and how to request an increase.

Console Announcements

Information about the announcements that occasionally appear in the Oracle Cloud Infrastructure Console

Prerequisites for Oracle Platform Services on Oracle Cloud Infrastructure

Instructions for setting up the resources required when running an Oracle Platform Service on Oracle Cloud Infrastructure.

Billing and Payment Tools Overview

Information about billing and payment tools that you can use to analyze your service usage and manage your costs.

My Services Use Cases

Use cases for the Oracle Cloud My Services API, to help you interact programmatically with My Services.

Security Credentials

This section describes the types of credentials you'll use when working with Oracle Cloud Infrastructure.

Console Password

- **What it's for:** Using the Console.
- **Format:** Typical password text string.
- **How to get one:** An administrator will provide you with a one-time password.
- **How to use it:** Sign in to the Console the first time with the one-time password, and then change it when prompted. Requirements for the password are displayed there. The one-time password expires in seven days. If you want to change the password later, see To change your Console password. Also, you or an administrator can reset the password in the Console or with the API (see To create or reset another user's Console password). Resetting the password creates a new one-time password that you'll be

prompted to change the next time you sign in to the Console. If you're blocked from signing in to the Console because you've tried 10 times in a row unsuccessfully, contact your administrator.

- **Note for Federated Users:** Federated users do not use a Console password. Instead, they sign in to the Console through their identity provider.

API Signing Key

- **What it's for:** Using the API (see Software Development Kits and Command Line Interface and Request Signatures).
- **Format:** RSA key pair in PEM format (minimum 2048 bits required).
- **How to get one:** See Required Keys and OCIDs.
- **How to use it:** In the Console, copy and paste the contents of the PEM public key file from the key pair (see How to Upload the Public Key). Then use the private key with the SDK or with your own client to sign your API requests. Note that after you've uploaded your first API key in the Console, you can use the API to upload any additional ones you want to use. If you provide the wrong kind of key (for example, your instance SSH key, or a key that isn't at least 2048 bits), you'll get an `InvalidKey` error.
- **Example:** The PEM public key looks something like this:

```
-----BEGIN PUBLIC KEY-----

MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAoTFqF...
...

-----END PUBLIC KEY--
```

Instance SSH Key

- **What it's for:** Accessing a compute instance.
- **Format:** For RSA, DSS, or DSA: minimum 2048 bits recommended. For ECDSA: minimum 128 bits recommended.
- **How to get one:** See Creating a Key Pair.

- **How to use it:** When you launch an instance, provide the public key from the key pair.
- **Example:** An RSA public key looks something like this:

```
ssh-rsa AAAAB3BzaC1yc2EAAAADAQABAAABAQD9BRwrUiLDki6P0+jZhwsjS2muM...

... john.smith@example.com
```

Auth Token

- **What it's for:** Authenticating with third-party APIs that do not support Oracle Cloud Infrastructure's signature-based authentication. For example, use an auth token as your password with Swift clients.
- **Format:** Typical password text string.
- **How to get one:** See Working with Auth Tokens.
- **How to use it:** Usage depends on the service your are authenticating with. Typically, you authenticate with third-party APIs by providing your Oracle Cloud Infrastructure Console login, your auth token provided by Oracle, and your organization's Oracle tenant name.

Regions and Availability Domains

Oracle Cloud Infrastructure is hosted in regions and availability domains. A region is a localized geographic area, and an availability domain is one or more data centers located within a region. A region is composed of one or more availability domains. Most Oracle Cloud Infrastructure resources are either region-specific, such as a virtual cloud network, or availability domain-specific, such as a compute instance. Traffic between availability domains and between regions is encrypted.

Availability domains are isolated from each other, fault tolerant, and very unlikely to fail simultaneously. Because availability domains do not share infrastructure such as power or cooling, or the internal availability domain network, a failure at one availability domain within a region is unlikely to impact the availability of the others within the same region.

The availability domains within the same region are connected to each other by a low latency, high bandwidth network, which makes it possible for you to provide high-availability

connectivity to the internet and on-premises, and to build replicated systems in multiple availability domains for both high-availability and disaster recovery.

Oracle is adding multiple cloud regions around the world to provide local access to cloud resources for our customers. To accomplish this quickly, we've chosen to launch regions in new geographies with one availability domain.

As regions require expansion, we have the option to add capacity to existing availability domains, to add additional availability domains to an existing region, or to build a new region. The expansion approach in a particular scenario is based on customer requirements as well as considerations of regional demand patterns and resource availability.

For any region with one availability domain, a second availability domain or region in the same country or geo-political area will be made available within a year to enable further options for disaster recovery that support customer requirements for data residency where they exist.

Regions are completely independent of other regions and can be separated by vast distances—across countries or even continents. Generally, you would deploy an application in the region where it is most heavily used, because using nearby resources is faster than using distant resources. However, you can also deploy applications in different regions for these reasons:

- To mitigate the risk of region-wide events such as large weather systems or earthquakes.
- To meet varying requirements for legal jurisdictions, tax domains, and other business or social criteria.

Region Name	Region Location	Region Key	Availability Domains
ap-seoul-1	Seoul, South Korea	ICN	1
ap-tokyo-1	Tokyo, Japan	NRT	1
ca-toronto-1	Toronto, Canada	YYZ	1

Region Name	Region Location	Region Key	Availability Domains
eu-frankfurt-1	Frankfurt, Germany	FRA	3
uk-london-1	London, United Kingdom	LHR	3
us-ashburn-1	Ashburn, VA	IAD	3
us-phoenix-1	Phoenix, AZ	PHX	3

To subscribe to a region, see Managing Regions.



Note

Your Tenancy's Availability Domain Names

Oracle Cloud Infrastructure randomizes the availability domains by tenancy to help balance capacity in the data centers. For example, the availability domain labeled PHX-AD-1 for tenancyA may be a different data center than the one labeled PHX-AD-1 for tenancyB. To keep track of which availability domain corresponds to which data center for each tenancy, Oracle Cloud Infrastructure uses tenancy-specific prefixes for the availability domain names. For example: the availability domains for your tenancy are something like *Uocm:*PHX-AD-1, *Uocm:*PHX-AD-2, and so on.

To get the specific names of your tenancy's availability domains, use the ListAvailabilityDomains operation, which is available in the IAM API. You can also see the names when you use the Console to launch an instance and choose which availability domain to launch the instance into.

Fault Domains

A fault domain is a grouping of hardware and infrastructure within an availability domain. Each availability domain contains three fault domains. Fault domains let you distribute your instances so that they are not on the same physical hardware within a single availability domain. A hardware failure or Compute hardware maintenance that affects one fault domain does not affect instances in other fault domains.

To control the placement of your compute, bare metal DB system, or virtual machine DB system instances, you can optionally specify the fault domain for a new instance at launch time. If you do not specify the fault domain, the system selects one for you. To change the fault domain for an instance, terminate it and launch a new instance in the preferred fault domain.

Use fault domains to:

- Protect against unexpected hardware failures
- Protect against planned outages due to Compute hardware maintenance
- See Fault Domains in Best Practices for Your Compute Instance for recommendations on using fault domains when provisioning application and database servers.
- See Fault Domain Considerations for 2-node Virtual Machine DB Systems and Availability Domain and Fault Domain Considerations for Data Guard for more information on using fault domains when provisioning Oracle bare metal and virtual machine DB systems.

Service Availability Across Regions

All Oracle Cloud Infrastructure regions offer core infrastructure services, including the following:

- Compute: Compute (Intel based Bare Metal & VM, DenseIO & Standard), Container Engine for Kubernetes, Registry
- Storage: Block Volume, File Storage, Object Storage, Archive Storage

- Networking: Virtual Cloud Network, Load Balancing, FastConnect (specific partners as available and requested)
- Database: Database, Exadata Cloud Service, Autonomous Data Warehouse, Autonomous Transaction Processing
- Edge: DNS
- Platform: Identity and Access Management, Tagging, Audit

Generally available cloud services beyond those in the preceding list are made available based on regional customer demand. Any service can be made available within a maximum of three months, with many services deploying more quickly. New cloud services are made available in regions as quickly as possible based on a variety of considerations including regional customer demand, ability to achieve regulatory compliance where applicable, resource availability, and other factors. Because of our low latency interconnect backbone, customers can use cloud services in other geographic regions with effective results when they are not available in their home region, provided that data residency requirements do not prevent them from doing so. We regularly work with customers to help ensure effective access to required services.

Resource Availability

The following sections list the resource types based on their availability: global across regions, within a single region, or within a single availability domain.



Tip

In general: IAM resources are global. DB Systems, instances, and volumes are specific to an availability domain. Everything else is regional. Exception: Subnets were originally designed to be specific to an availability domain. Now, you can create regional subnets, which is what Oracle recommends.

Global Resources

- API signing keys
- compartments
- dynamic groups
- federation resources
- groups
- policies
- tag namespaces
- tag keys
- users

Regional Resources

- alarms
- buckets: Although buckets are regional resources, they can be accessed from any location if you use the correct region-specific Object Storage URL for the API calls.
- clusters
- customer-premises equipment (CPE)
- DHCP options sets
- dynamic routing gateways (DRGs)
- encryption keys
- images
- internet gateways
- jobs
- key vaults
- load balancers
- local peering gateways (LPGs)

- metrics
- NAT gateways
- node pools
- ons-subscriptions
- ons-topics
- repositories
- reserved public IPs
- route tables
- security lists
- service gateways
- stacks
- subnets: When you create a subnet, you choose whether it's regional or specific to an availability domain. Oracle recommends using regional subnets.
- virtual cloud networks (VCNs)
- volume backups: They can be restored as new volumes to any availability domain within the same region in which they are stored.

Availability Domain-Specific Resources

- DB Systems
- ephemeral public IPs
- instances: They can be attached only to volumes in the same availability domain.
- subnets: When you create a subnet, you choose whether it is regional or specific to an availability domain. Oracle recommends using regional subnets.
- volumes: They can be attached only to an instance in the same availability domain.

IP Address Ranges

This section lists public IP address ranges for services that are deployed in Oracle Cloud Infrastructure. You should whitelist these CIDR blocks to ensure access to the services.

Public IP Address Ranges for VCNs

Each VCN's public IP addresses can come from the following CIDR blocks.

ap-seoul-1

- 132.145.80.0/21

ap-tokyo-1

- 132.145.112.0/21

ca-toronto-1

- 132.145.96.0/21
- 132.145.104.0/22

eu-frankfurt-1

- 130.61.8.0/21
- 130.61.16.0/20
- 130.61.32.0/19
- 130.61.64.0/19
- 130.61.96.0/23
- 130.61.104.0/21

- 130.61.112.0/20
- 132.145.232.0/21
- 132.145.240.0/21
- 132.145.248.0/21

uk-london-1

- 132.145.8.0/21
- 132.145.16.0/20
- 132.145.32.0/19
- 132.145.64.0/23

us-ashburn-1

- 129.213.8.0/21
- 129.213.16.0/20
- 129.213.32.0/19
- 129.213.64.0/18
- 129.213.128.0/18
- 129.213.192.0/21
- 129.213.208.0/21
- 132.145.128.0/18
- 132.145.192.0/19

us-phoenix-1

- 129.146.0.0/21
- 129.146.8.0/22

- 129.146.16.0/20
- 129.146.32.0/19
- 129.146.64.0/18
- 129.146.128.0/19
- 129.146.160.0/20
- 129.146.208.0/21

Public IP Address Ranges for the Oracle Services Network

The *Oracle Services Network* is a conceptual network in Oracle Cloud Infrastructure that is reserved for Oracle services. Following are the CIDR blocks of the Oracle Services Network in each region. This list is not expected to change frequently because the blocks were sized to accommodate additional future services.

ap-seoul-1

- 134.70.96.0/22
- 140.91.40.0/23
- 140.204.24.128/25
- 192.29.16.0/21

ap-tokyo-1

- 134.70.80.0/22
- 140.91.32.0/23
- 140.204.8.128/25
- 192.29.32.0/21

ca-toronto-1

- 134.70.72.0/22
- 140.91.28.0/23
- 140.204.0.128/25
- 192.29.0.0/21
- 192.29.8.0/22

eu-frankfurt-1

- 130.61.0.0/21
- 134.70.40.0/21
- 134.70.48.0/22
- 138.1.0.0/20
- 138.1.40.0/21
- 138.1.64.0/20
- 138.1.96.0/21
- 138.1.104.0/22
- 138.1.160.0/19
- 138.1.192.0/20
- 140.91.16.0/22
- 140.91.20.0/23
- 140.91.198.0/23
- 147.154.128.0/18
- 147.154.192.0/20
- 147.154.208.0/21

uk-london-1

- 130.35.112.0/22
- 132.145.0.0/21
- 134.70.56.0/21
- 134.70.64.0/22
- 138.1.16.0/20
- 138.1.80.0/20
- 138.1.208.0/20
- 138.1.224.0/19
- 140.91.22.0/23
- 140.91.24.0/22
- 140.91.200.0/23
- 147.154.224.0/19

us-ashburn-1

- 129.213.0.0/21
- 130.35.16.0/20
- 130.35.48.0/20
- 130.35.64.0/19
- 130.35.96.0/20
- 130.35.120.0/21
- 130.35.144.0/20
- 130.35.176.0/20
- 130.35.192.0/19
- 130.35.224.0/22

- 130.35.232.0/21
- 134.70.24.0/21
- 134.70.32.0/22
- 138.1.48.0/21
- 140.91.10.0/23
- 140.91.12.0/22
- 140.91.196.0/23
- 147.154.0.0/18
- 147.154.64.0/20
- 147.154.80.0/21

us-phoenix-1

- 129.146.12.0/22
- 130.35.0.0/20
- 130.35.128.0/20
- 130.35.240.0/20
- 134.70.8.0/21
- 134.70.16.0/22
- 138.1.32.0/21
- 138.1.128.0/19
- 140.91.4.0/22
- 140.91.8.0/23
- 140.91.194.0/23
- 147.154.96.0/19

Public IP Addresses for the Oracle YUM Repos

The Oracle YUM repos have regional public endpoints that follow this format:

yum-*<region>*.oracle.com

For example:

- yum-ashburn.oracle.com (for region us-ashburn-1)
- yum-frankfurt.oracle.com (for region eu-frankfurt-1)

You can use DNS lookup to determine the public IP address for each endpoint.

Resource Identifiers

This chapter describes the different ways your Oracle Cloud Infrastructure resources are identified.

Oracle Cloud IDs (OCIDs)

Most types of Oracle Cloud Infrastructure resources have an Oracle-assigned unique ID called an *Oracle Cloud Identifier* (OCID). It's included as part of the resource's information in both the Console and API.



Important

To use the API, you need the OCID for your tenancy. For information about where to find it, see the next section.

OCIDs use this syntax:

```
ocid1.<RESOURCE TYPE>.<REALM>.[REGION][.FUTURE USE].<UNIQUE ID>
```

- **ocid1:** The literal string indicating the version of the OCID.
- **resource type:** The type of resource (for example, `instance`, `volume`, `vcn`, `subnet`, `user`, `group`, and so on).
- **realm:** The realm the resource is in. A *realm* is a set of regions that share entities. The only possible value is `oc1`.
- **region**: The region the resource is in (for example, `phx`, `iad`, `eu-frankfurt-1`). With the introduction of the Frankfurt region, the format switched from a three-character code to a longer string. This part is present in the OCID only for regional resources or those specific to a single availability domain. If the region is not applicable to the resource, this part might be blank (see the example tenancy ID below).
- **future use:** Reserved for future use. Currently blank.
- **unique ID:** The unique portion of the ID. The format may vary depending on the type of resource or service.

Example OCIDs

Tenancy:

```
ocid1.tenancy.oc1..aaaaaaaaba3pv6wkcr4jqae5f44n2b2m2yt2j6rx32uzr4h25vqstifsfdsq
```

Instance:

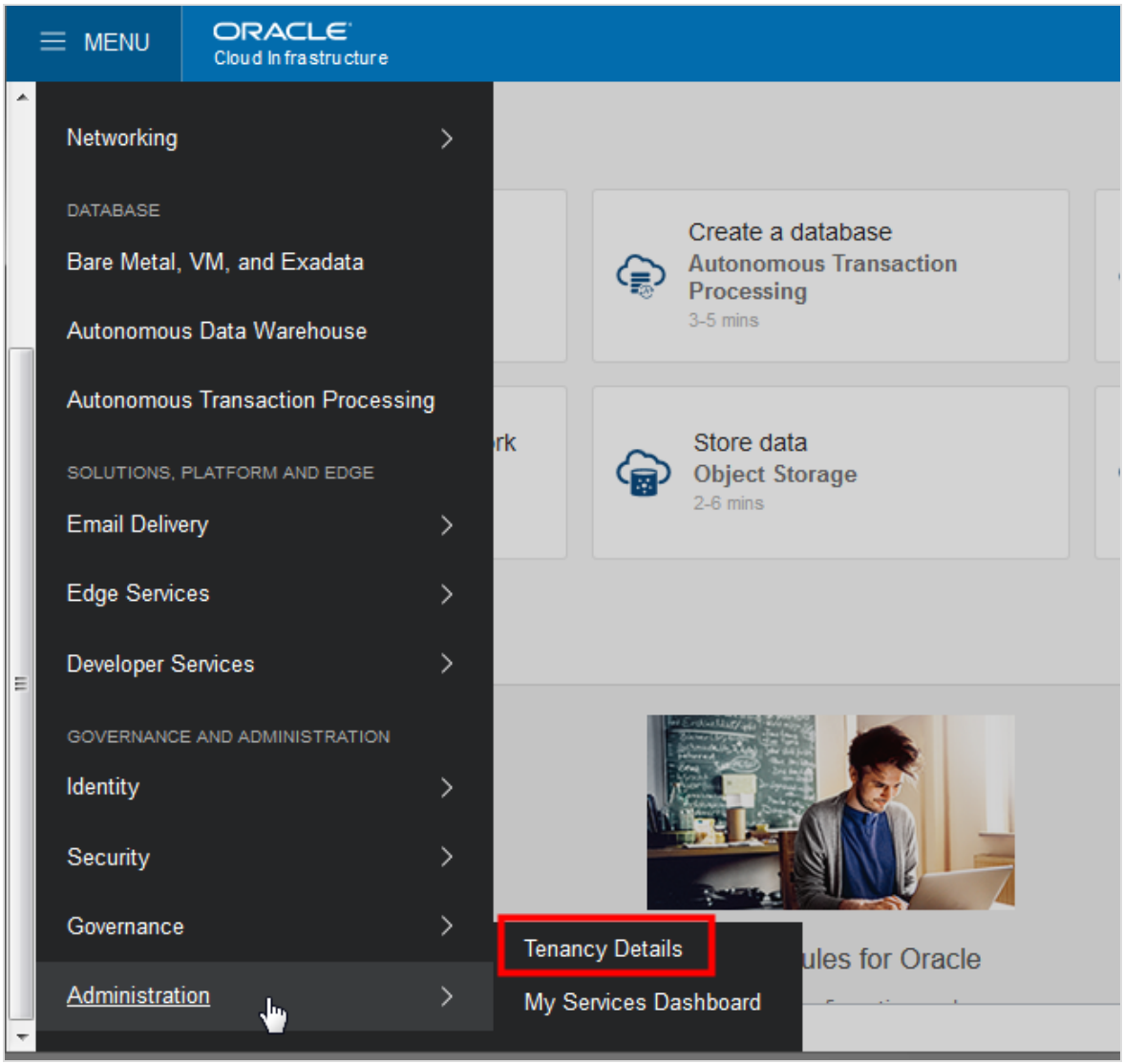
```
ocid1.instance.oc1.phx.abuw4ljrlsfiqw6vzzxb43vyypt4pkodawglp3wqxjqofakrwvou52gb6s5a
```

Where to Find Your Tenancy's OCID

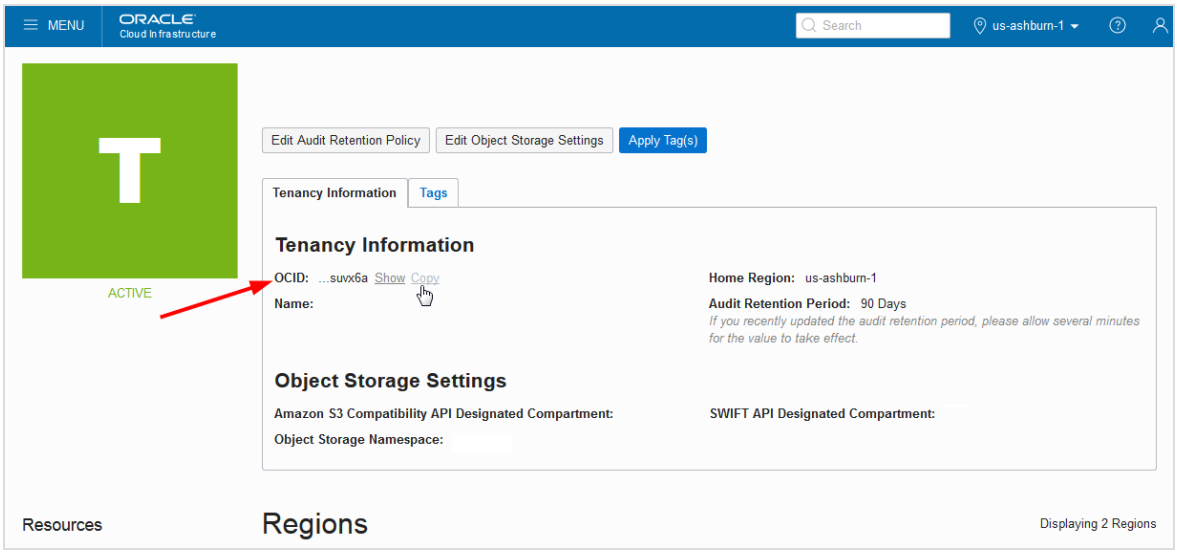
If you use the Oracle Cloud Infrastructure API, you need your tenancy's OCID in order to sign the API requests. You also use the tenancy ID in some of the IAM API operations.

Get the tenancy OCID from the Oracle Cloud Infrastructure Console on the **Tenancy Details** page:

1. Open the navigation menu, under Governance and Administration, go to **Administration** and click **Tenancy Details**.



2. The tenancy OCID is shown under **Tenancy Information**. Click **Copy** to copy it to your clipboard.



The tenancy OCID looks something like this (notice the word "tenancy" in it):

```
ocid1.tenancy.oc1..<unique_ID>
```

Name and Description (IAM Only)

The IAM service requires you to assign a unique, unchangeable *name* to each of your IAM resources (users, groups, policies, and compartments). The name must be unique within the scope of the type of resource (for example, you can only have one user with the name BobSmith). Notice that this requirement is specific to IAM and not the other services.

The name you assign to a user at creation is their login for the Console.

You can use these names instead of the OCID when writing a policy (for example, `Allow group <GROUP NAME> to manage all-resources in compartment <COMPARTMENT NAME>`).

In addition to the name, you must also assign a *description* to each of your IAM resources (although it can be an empty string). It can be a friendly description or other information that helps you easily identify the resource. The description does not have to be unique, and you can change it whenever you like. For example, you might want to use the description to store

the user's email address if you're not already using the email address as the user's unique name.

Display Name

For most of the Oracle Cloud Infrastructure resources you create (other than those in IAM), you can optionally assign a *display name*. It can be a friendly description or other information that helps you easily identify the resource. The display name does not have to be unique, and you can change it whenever you like. The Console shows the resource's display name along with its OCID.

Resource Tags

When you have many resources (for example, instances, VCNs, load balancers, and block volumes) across multiple compartments in your tenancy, it can become difficult to track resources used for specific purposes, or to aggregate them, report on them, or take bulk actions on them. *Tagging* allows you to define keys and values and associate them with resources. You can then use the tags to help you organize and list resources based on your business needs.

There are two types of tags:

Defined tags are set up in your tenancy by an administrator. Only users granted permission to work with the defined tags can apply them to resources.

Free-form tags can be applied by any user with permissions on the resource.

For more detailed information about tags and their features, see Overview of Tags.



Tip

Watch a video to introduce you to the concepts and features of tagging: Introduction to Tagging.

Working with Resource Tags

Not all resources support tags. See Taggable Resources for the list of resources that support tags.

To add a defined tag to an existing resource

To apply a defined tag, you must have permission to use the namespace.

1. Open the Console, go to the details page of the resource you want to tag.
   For example, to tag a compute instance: Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**. A list of the instances in your current compartment is displayed. Find the instance that you want to tag, and click its name to view its details page.
2. Click **Apply Tags**.
3. In the **Apply Tags to the Resource** dialog:
   a. Select the **Tag Namespace**.
   b. Select the **Tag Key**.
   c. Enter a **Value**.
   d. To apply another tag, click **+ Additional Tag**.
   e. When finished adding tags, click **Apply Tag(s)**.

To add a free-form tag to an existing resource

1. Open the Console, go to the details page of the resource you want to tag.
   For example, to tag a compute instance: Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**. A list of the instances in your current compartment is displayed. Find the instance that you want to tag, and click its name to view its details page.
2. Click **Apply Tags**.

3. In the **Apply Tags to the Resource** dialog:
    a. Select **None (apply a free-form tag)**.
    b. Enter the **Tag Key**.
    c. Enter a **Value**.
    d. To apply another tag, click **+ Additional Tag**.
    e. When finished adding tags, click **Apply Tag(s)**.

To add a tag during resource creation

You can apply tags during resource creation. The location of the **Apply Tag(s)** option in the dialog varies by resource. The general steps are:

1. In the resource Create dialog, click **Apply Tags**.
2. In the **Apply Tags to the Resource** dialog:
    a. Select the **Tag Namespace**, or select **None** to apply a free-form tag.
    b. Select or enter the **Tag Key**.
    c. Enter a **Value**.
    d. To apply another tag, click **+ Additional Tag**.
    e. Click **Apply Tag(s)**.

To filter a list of resources by a tag

Open the Console, click the service name and then click the resource you want to view. The left side of the page shows all the filters currently applied to the list.

For example, to view compute instances: Click **Compute** and then click **Instances**, to see the list of instances in your current compartment.

To filter a list of resources by a defined tag

1. Next to **Tag Filters**, click **add**.
2. In the **Apply a Tag Filter** dialog, enter the following:
    a. **With Tag Type:** Select **Defined Tag**.
    b. **In Namespace:** Select the tag namespace.
    c. **With Key:** Select a specific key.
    d. **With Value:** Select from the following:
        - Any - returns all resources tagged with the selected namespace and key, regardless of the tag value.
        - Equal to - returns resources with the tag value you enter in the text box. Enter a single value in the text box. To specify multiple values for the same namespace and key, click **OR** to display another text box. Enter one value per text box.

To filter a list of resources by a free-form tag

1. Next to **Tag Filters**, click **add**.
2. In the **Apply a Tag Filter** dialog, enter the following:
    a. **With Tag Type:** Select **Freeform Tag**.
    b. **With Key:** Enter the tag key.
    c. **With Value:** Select from the following:
        - Any - returns all resources tagged with the selected free-form tag key, regardless of the tag value.
        - Equal to - returns resources with the tag value you enter in the text box. Enter a single value in the text box. To specify multiple values for the same key, click **OR** to display another text box. Enter one value per text box.

To update a tag applied to a resource

1. Open the Console, click the service name and then click the resource you want to view. For example, to view compute instances: Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**. A list of the instances in your current compartment is displayed. Find the instance that you want to update, and click its name to view its details page.
2. Click **Tags**.
   The list of tags applied to the resource is displayed.
3. Find the tag you want to update and click the pencil icon next to it.
4. Enter the new value.
5. Click **Save**.

To remove a tag from a resource

1. Open the Console, click the service name and then click the resource you want to view. For example, to view a compute instance: Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**. A list of the instances in your current compartment is displayed. Find the instance that you want to remove the tag from, and click its name to view its details page.
2. Click **Tags**.
   The list of tags applied to the resource is displayed.
3. Find the tag you want to remove and click the pencil icon next to it.
4. Click **Remove Tag**.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

To apply a tag to a resource using the API, use the appropriate resource's `create` or `update` operation.

Resource Monitoring

You can monitor the health, capacity, and performance of your Oracle Cloud Infrastructure resources when needed using queries or on a passive basis using alarms. Queries and alarms rely on metrics emitted by your resource to the Monitoring service.

Prerequisites

- IAM policies: To monitor resources, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. The policy must give you access to the monitoring services as well as the resources being monitored. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in. For more information on user authorizations for monitoring, see the Authentication and Authorization section for the related service: Monitoring or Notifications.
- Metrics exist in Monitoring: The resources that you want to monitor must emit metrics to the Monitoring service.
- Compute instances: To emit metrics, Compute instances must be monitoring-enabled. OracleCloudAgent software installation may also be required. For more information, see Enabling Monitoring for Compute Instances.

Working with Resource Monitoring

Not all resources support monitoring. See Supported Services for the list of resources that support the Monitoring service, which is required for queries and alarms used in monitoring.

The Monitoring service works with the Notifications service to notify you when metrics breach. For more information about these services, see Monitoring Overview and Notifications Overview.

To view default metric charts for a resource

On the page for the resource of interest, under **Resources**, click **Metrics**.

For example, to view metric data for a Compute instance:

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
2. Click the instance you're interested in.
3. On the instance detail page, under **Resources**, click **Metrics**.
   A chart is shown for each metric. For a list of metrics related to Compute instances, see Compute Metrics.

The Console displays the last hour of metric data for the selected resource. A chart is shown for each metric emitted by the selected resource.

For a list of metrics emitted by your resource, see Supported Services.

To view default metric charts for a set of resources

Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Service Metrics**.

The **Service Metrics** page displays the default charts for all resources in the first accessible **Compartment** and **Metric Namespace**. Very small or large values are indicated by International System of Units (SI units), such as M for mega (10 to the sixth power).

Don't see all expected resources or metrics?

- Try a different time range.

- Confirm that the missing resources are emitting metrics. See Enabling Monitoring for Compute Instances.
- Review limits information. Limits information for returned data includes the 100,000 data point maximum and time range maximums (determined by resolution, which relates to interval). See MetricData Reference.

To create a query

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Metrics Explorer**.
   The **Metrics Explorer** page displays an empty chart with fields to build a query.
2. Fill in the fields for a new query.
   - **Compartment**: The compartment containing the resources that you want to monitor. By default, the first accessible compartment is selected.
   - **Metric Namespace**: The service or application emitting metrics for the resources that you want to monitor.
   - **Metric Name**: The name of the metric. Only one metric can be specified. Metric selections depend on the selected compartment and metric namespace.
     Example: **CpuUtilization**
   - **Interval**: The aggregation window.

     ### Interval values

     - **1m** - 1 minute
     - **5m** - 5 minutes
     - **1h** - 1 hour



Note

For metric queries, the interval you select drives the default resolution of the request, which determines the maximum time range of data returned.

Maximum time range returned for a query

The maximum time range returned for a metric query depends on the resolution. By default, for metric queries, the resolution is the same as the query interval. Following are the maximum time ranges returned for each interval selection available in the Console.

Interval	Default resolution (metric queries)	Maximum time range returned
1h	1 hour	90 days
5m	5 minutes	30 days
1m	1 minute	7 days



For more information about the resolution parameter as used in metric queries, see SummarizeMetricsData.

- **Statistic**: The aggregation function.

  Statistic values

  - **COUNT**- The number of observations received in the specified time period.
  - **MAX** - The highest value observed during the specified time period.
  - **MEAN** - The value of Sum divided by Count during the specified time period.
  - **MIN** - The lowest value observed during the specified time period.
  - **P50** - The value of the 50th percentile.
  - **P90** - The value of the 90th percentile.
  - **P95** - The value of the 95th percentile.
  - **P99** - The value of the 99th percentile.
  - **P99.5** - The value of the 99.5th percentile.
  - **RATE** - The per-interval average rate of change.
  - **SUM** - All values added together.

- **Metric dimensions**: Optional filters to narrow the metric data evaluated.

  Dimension fields

  - **Dimension Name**: A qualifier specified in the metric definition. For example, the dimension `resourceId` is specified in the metric definition for `CpuUtilization`.

    Note

    Long lists of dimensions are trimmed.

    - To view dimensions by name, type one or more characters in the box. A refreshed (trimmed) list shows matching dimension names.
    - To retrieve all dimensions for a given metric, use the following API operation: ListMetrics

  - **Dimension Value**: The value you want to use for the specified dimension. For example, the resource identifier for your instance of interest.
  - **+ Additional dimension**: Adds another name-value pair for a dimension.

- **Aggregate Metric Streams**: Aggregates all results to plot a single aggregated average for all metric streams. This average is plotted as a single line on the metric chart. This operation is helpful when you want to plot a metric as one line for all resources.

3. Click **Update Chart**.
   The chart shows the results of your new query. Very small or large values are indicated by International System of Units (SI units), such as M for mega (10 to the sixth power). Units correspond to the selected metric and do not change by statistic.

   ### Troubleshooting Errors and Query Limits

   If you see an error that the query has exceeded the maximum number of metric streams, then update the query to evaluate a number of metric streams that is within the limit. For example, you can reduce the metric streams by specifying dimensions. You can continue to evaluate all metric streams that were in the original query by spreading the metric streams across multiple queries (or alarms).
   Limits information for returned data includes the 100,000 data point maximum and time range maximums (determined by resolution, which relates to interval). See MetricData Reference.

4. To customize the y-axis label or range, type the label you want into **Y-Axis Label** or type the minimum and maximum values you want into **Y-Axis Min Value** and **Y-Axis Max Value**.
   Only numeric characters are allowed for custom ranges. Custom labels and ranges are not persisted in shared queries (MQL).
5. To view the query as a Monitoring Query Language (MQL) expression, click **Advanced Mode**.
   Advanced Mode is located on the right, under the chart.
   Use Advanced Mode to edit your query using MQL syntax to aggregate results by group. The MQL syntax also supports additional parameter values. For more information about query parameters in Basic Mode and Advanced Mode, see Monitoring Query Language (MQL) Reference.
6. To create another query, click **Add Query** below the chart.

To create an alarm

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Alarm Definitions**.
2. Click **Create alarm**.



Note

You can also create an alarm from a predefined query on the **Service Metrics** page. Expand **Options** and click **Create an Alarm on this Query**. For more information about service metrics, see Viewing Default Metric Charts.

3. On the **Create Alarm** page, under **Define alarm**, fill in or update the alarm settings:



Note

To toggle between Basic Mode and Advanced Mode, click **Switch to Advanced Mode** or **Switch to Basic Mode** (to the right of **Define Alarm**).

Basic Mode (default)

By default, this page uses **Basic Mode**, which separates the metric from its dimensions and its trigger rule.

- **Alarm Name**: User-friendly name for the new alarm.
- **Alarm Severity**: The perceived type of response required when the alarm is in the firing state.

- **Alarm Body**: The human-readable content of the notification delivered. Oracle recommends providing guidance to operators for resolving the alarm condition. Consider adding links to standard runbook practices. Example: "High CPU usage alert. Follow runbook instructions for resolution."
- **Tags (optional):** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
- **Metric description**: The metric to evaluate for the alarm condition.
    - **Compartment**: The compartment containing the resources that emit the metrics evaluated by the alarm. The selected compartment is also the storage location of the alarm. By default, the first accessible compartment is selected.
    - **Metric Namespace**: The service or application emitting metrics for the resources that you want to monitor.
    - **Metric Name**: The name of the metric. Only one metric can be specified. Example: **CpuUtilization**

- **Interval**: The aggregation window, or the frequency at which data points are aggregated.

  ### Interval values

  - **1m** - 1 minute
  - **5m** - 5 minutes
  - **1h** - 1 hour

  Note

  For alarm queries, the specified interval has no effect on the resolution of the request. The only valid value of the resolution for an alarm query request is `1m`. For more information about the resolution parameter as used in alarm queries, see Alarm.

- **Statistic**: The aggregation function.

  ### Statistic values

  - **COUNT**- The number of observations received in the specified time period.
  - **MAX** - The highest value observed during the specified time period.
  - **MEAN** - The value of Sum divided by Count during the specified time period.
  - **MIN** - The lowest value observed during the specified time period.

- **P50** - The value of the 50th percentile.
- **P90** - The value of the 90th percentile.
- **P95** - The value of the 95th percentile.
- **P99** - The value of the 99th percentile.
- **P99.5** - The value of the 99.5th percentile.
- **RATE** - The per-interval average rate of change.
- **SUM** - All values added together.

- **Metric dimensions**: Optional filters to narrow the metric data evaluated.

  Dimension fields

  - **Dimension Name**: A qualifier specified in the metric definition. For example, the dimension `resourceId` is specified in the metric definition for `CpuUtilization`.

    Note

    Long lists of dimensions are trimmed.

    - To view dimensions by name, type one or more characters in the box. A refreshed (trimmed) list shows matching dimension names.
    - To retrieve all dimensions for a given metric, use the following API operation: ListMetrics

  - **Dimension Value**: The value you want to use for the specified dimension. For example, the resource identifier for your instance of interest.
  - **+ Additional dimension**: Adds another name-value pair for a dimension.

- **Trigger rule**: The condition that must be satisfied for the alarm to be in the firing state. The condition can specify a threshold, such as 90% for CPU Utilization, or an absence.

- **Operator**: The operator used in the condition threshold.

  Operator values

  - **greater than**
  - **greater than or equal to**
  - **equal to**
  - **less than**
  - **less than or equal to**
  - **between** (inclusive of specified values)
  - **outside** (inclusive of specified values)
  - **absent**

- **Value**: The value to use for the condition threshold.
- **Trigger Delay Minutes**: The number of minutes that the condition must be maintained before the alarm is in firing state.

Advanced Mode

Click **Advanced Mode** or **Switch to Advanced Mode** to view the alarm query as a Monitoring Query Language (MQL) expression. Edit your query using MQL syntax to aggregate results by group or for additional parameter values. See Monitoring Query Language (MQL) Reference.

- **Alarm Name**: User-friendly name for the new alarm.
- **Alarm Severity**: The perceived type of response required when the alarm is in the firing state.
- **Alarm Body**: The human-readable content of the notification delivered. Oracle recommends providing guidance to operators for resolving the alarm condition.

Consider adding links to standard runbook practices. Example: "High CPU usage alert. Follow runbook instructions for resolution."

- **Tags (optional):** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
- **Metric description, dimensions, and trigger rule**: The metric to evaluate for the alarm condition, including dimensions and the trigger rule.
  - **Compartment**: The compartment containing the resources that emit the metrics evaluated by the alarm. The selected compartment is also the storage location of the alarm. By default, the first accessible compartment is selected.
  - **Metric Namespace**: The service or application emitting metrics for the resources that you want to monitor.
  - **Query Code Editor** box: The alarm query as a Monitoring Query Language (MQL) expression.
    Example alarm query:

    ```
    CpuUtilization[1m]{availabilityDomain=AD1}.groupBy(poolId).percentile(0.9) > 85
    ```

    For query syntax and examples, see Working with Metric Queries.
  - **Trigger Delay Minutes**: The number of minutes that the condition must be maintained before the alarm is in firing state.

The chart below the **Define alarm** section dynamically displays the last six hours of emitted metrics according to currently selected fields for the query. Very small or large values are indicated by International System of Units (SI units), such as M for mega (10 to the sixth power).

4. Set up notifications: Under **Notifications**, fill in the fields.

- **Destinations**:
  - **Destination Service**: The provider of the destination to use for notifications.

    Available options:
    - **Notifications Service**.
  - **Compartment**: The compartment storing the topic to be used for notifications. Can be a different compartment from the alarm and metric. By default, the first accessible compartment is selected.
  - **Topic**: The topic to use for notifications. Each topic supports a subscription protocol, such as PagerDuty.
  - **Create a topic**: Sets up a topic and subscription protocol in the selected compartment, using the specified destination service.
    - **Topic Name**: User-friendly name for the new topic. Example: "Operations Team " for a topic used to notify operations staff of firing alarms.
    - **Topic Description**: Description of the new topic.
    - **Subscription Protocol**: Medium of communication to use for the new topic. Configure your subscription for the protocol you want:

      Email subscription
      - **Subscription protocol**: Select **Email**.
      - **Email Addresses**: Type an email address.

      PagerDuty subscription
      - **Subscription Protocol**: Select **HTTPS (PagerDuty)**.

- **PagerDuty Integration URL**: Type (or copy and paste) the *integration key* portion of the URL for your PagerDuty subscription. (The other portions of the URL are hard-coded.)
  For information on setting up and retrieving your integration key, see the PagerDuty documentation.
  Example integration key: Your68a4Integration4dd8Key12f4f6
  Example integration URL: https://events.pagerduty.com/integration/Your68a4Integration4dd8Key12f4f6/enqueue

◦ **+ Additional destination service**: Adds another destination service and topic to use for notifications.



Note

Each alarm is limited to one destination per supported destination service.

- **Repeat Notification?**: While the alarm is in the firing state, resends notifications at the specified interval.
- **Notification Interval**: The period of time to wait before resending the notification.
- **Suppress Notifications**: Sets up a suppression time window during which to suspend evaluations and notifications. Useful for avoiding alarm notifications during system maintenance periods.
  - **Suppression Description**
  - **Start Time**
  - **End Time**

5. If you want to disable the new alarm, clear **Enable This Alarm?**.

6. Click **Save alarm**.
   The new alarm is listed on the **Alarm Definitions** page.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

To create a query, use the SummarizeMetricsData operation.

To create an alarm, use the CreateAlarm operation.

Service Limits

This topic describes the service limits for Oracle Cloud Infrastructure and the process for requesting a service limit increase.

About Service Limits and Usage

When you sign up for Oracle Cloud Infrastructure, a set of service limits are configured for your tenancy. The service limit is the quota or allowance set on a resource. For example, your tenancy is allowed a maximum number of compute instances per availability domain. These limits are generally established with your Oracle sales representative when you purchase Oracle Cloud Infrastructure. If you did not establish limits with your Oracle sales representative, or, if you signed up through the Oracle Store, default or trial limits are set for your tenancy. You can request to have a service limit raised.

You can view your tenancy's limits and usage by region in the Console. Be aware that:

- The Console may not yet display limits and usage information for all of the Oracle Cloud Infrastructure services or resources.
- The usage level listed for a given resource type could be greater than the limit if the limit was reduced after the resources were created.

- If all the resource limits are listed as 0, this means your account has been suspended. For help, contact Oracle Support.

If you don't yet have a tenancy or a user login for the Console, or if you don't find a particular limit listed in the Console, see Limits by Service for the default tenancy limits.

To view your tenancy's limits and usage (by region)



Note

Required Permission

If you're in the Administrators group, you have permission to view the limits and usage. If you're not, here's an example IAM policy that grants the required permission to users in a group called LimitsAndUsageViewers:

```
Allow group LimitsAndUsageViewers to inspect limits in tenancy
```

1. Open the Console. Open the **User** menu () and click **Tenancy:** *<your_tenancy_name>*.
2. Click **Service Limits** on the left side of the page.

Your resource limits and usage for the specific region are displayed, broken out by service. If a given resource type has limits per availability domain, the limit and usage for each availability domain is displayed.

When You Reach a Service Limit

When you reach the service limit for a resource, you receive an error when you try to create a new resource of that type. You are then prompted to submit a request to increase your limit. You cannot create a new resource until you are granted an increase to your service limit or you terminate an existing resource. Note that service limits apply to a specific scope, and

when the service limit in one scope is reached you may still have resources available to you in other scopes (for example, other availability domains).

Requesting a Service Limit Increase

You can submit a request to increase your service limits from within the Console. If you try to create a resource for which limit has been met, you'll be prompted to submit a limit increase request. Additionally, you can launch the request from the service limits page or at any time by clicking the link under the **Help** menu (?).

Note that the service limit increase is not immediate.

To request a service limit increase

1. Open the **Help** menu (?), go to **Support** and click **Request service limit increase**.
2. Enter the following:
    - **Primary Contact Details:** Enter the name and email address of the person making the request. Enter one email address only. A confirmation will be sent to this address.
    - **Service Category:** Select the appropriate category for your request.
    - **Resource:** Select the appropriate resource.
      Depending on your selection for resource, additional fields might display for more specific information.
    - **Reason for Request:** Enter a reason for your request. If your request is urgent or unusual, please provide details here.
3. Click **Submit Request**.

After you submit the request, the request is reviewed. If your request is awarded, a confirmation email is sent to the address provided in the primary contact details.

If we need additional information about your request, a follow-up email is sent to the address provided in the primary contact details.

Limits by Service

The following tables list the default limits for each service. Note the scope that each limit applies to (for example, per availability domain, per region, per tenant, etc.).



Note

Some services have additional limits. For more information, see the overview of each service.

Block Volume Limits

Limits apply to each availability domain. There are three availability domains per region.

Resource	Monthly Universal Credits	Pay-as-You-Go or Promo
Block Volumes aggregated size	100 TB	30 TB
Backups	1000	500

Compute Limits

Limits apply to each availability domain, unless otherwise noted. There are three availability domains per region.

Bare Metal Servers

Resource	Monthly Universal Credits	Pay-as-You-Go or Promo
BM.Standard2.52	1 (52 cores)	Contact Us
BM.DenseIO2.52	1 (52 cores)	Contact Us
BM.GPU2.2	1 (28 cores) (us-ashburn-1, eu-frankfurt-1)	Contact Us
BM.GPU3.8	Contact Us	Contact Us
BM.HPC2.36	Contact Us	Contact Us
BM.Standard.E2.64	1 (64 cores) (us-ashburn-1, uk-london-1)	Contact Us
Custom Images	100 per region	25 per region

Virtual Machines

Resource	Monthly Universal Credits	Pay-as-You-Go or Promo
VM.Standard2.1	40	1
VM.Standard2.2	30	1
VM.Standard2.4	30	Contact Us

Resource	Monthly Universal Credits	Pay-as-You-Go or Promo
VM.Standard.E2.1	5 (us-ashburn-1, uk-london-1)	1 (us-ashburn-1, uk-london-1)
VM.Standard.E2.2	5 (us-ashburn-1, uk-london-1)	1 (us-ashburn-1, uk-london-1)
VM.Standard.E2.4	5 (us-ashburn-1, uk-london-1)	Contact Us
VM.Standard.E2.8	5 (us-ashburn-1, uk-london-1)	Contact Us
VM.Standard2.8	10 (us-phoenix-1, us-ashburn-1) 5 (other regions)	Contact Us
VM.Standard2.16	10 (us-phoenix-1, us-ashburn-1) 5 (other regions)	Contact Us
VM.Standard2.24	10 (us-phoenix-1, us-ashburn-1) 5 (other regions)	Contact Us
VMDenseIO2.8	10	Contact Us
VMDenseIO2.16	10	Contact Us
VMDenseIO2.24	10	Contact Us
VM.GPU3.1	Contact Us	Contact Us
VM.GPU3.2	Contact Us	Contact Us
VM.GPU3.4	Contact Us	Contact Us

Container Engine for Kubernetes Limits

Container Engine for Kubernetes limits are **regional**.

Resource	Monthly Universal Credits	Pay-as-You-Go or Promo
Clusters	3 clusters per OCI region	1 cluster per OCI region
Nodes	1000 nodes per cluster	1000 nodes per cluster

Data Transfer Limits

Data Transfer limits are **regional**.

Data Transfer Disk

Resource	Monthly Universal Credits	Pay-As-You-Go
Transfer package	Contact Us	Contact Us

File a service request at My Oracle Support to increase the service limits for Data Transfer Disk. See Requesting a Service Limit Increase for details.

Data Transfer Appliance

Resource	Monthly Universal Credits	Pay-As-You-Go
Transfer appliances	Contact Your CSM	Contact Your CSM

Contact your Oracle Account Manager or Customer Success Manager (CSM) to place an order for transfer appliances. Your service limits are automatically set to the number of transfer appliances ordered. You do not need to file a service request to increase the service limits for Data Transfer Appliance.

Database Limits

Database limits are per availability domain.

Resources	Monthly Flex	Pay-as-You-Go or Promo
Autonomous Data Warehouse - Total OCPUs	128 cores	8 cores
Autonomous Transaction Processing - Total OCPUs	128 cores	8 cores
VM.Standard1 -Total OCPUs	10 cores	2 cores
VM.Standard2 -Total OCPUs	100 cores (us-phoenix-1, us-ashburn-1) 50 cores (eu-frankfurt-1, uk-london-1)	2 cores
Total VM DB Block Storage (see note)	10TB	2TB
BM.DenseIO1.36 (see availability note)	1 instance	1 instance
BM.DenseIO2.52	1 instance	1 instance
BM.HighIO1.36	1 instance	1 instance
Exadata.Quarter1.84 - X6	Contact Us	not available
Exadata.Half1.168 - X6	Contact Us	not available
Exadata.Full1.336 - X6	Contact Us	not available

Resources	Monthly Flex	Pay-as-You-Go or Promo
Exadata.Quarter2.92 - X7	Contact Us	not available
Exadata.Half2.184 - X7	Contact Us	not available
Exadata.Full2.368 - X7	Contact Us	not available
BM.RACLocalStorage1.72	Contact Us	Contact Us



Note

Total VM DB Block Storage - covers block storage for all VM.Standard1 and VM.Standard2 virtual machine databases.

BM.DenseIO1.36 - This DB system shape is available only to monthly universal credit customers with tenancies existing on or before November 9th, 2018, in the us-phoenix-1, us-ashburn-1, and eu-frankfurt-1 regions.

DNS Limits

DNS limits are **global**.

Resources	Monthly Universal Credits	Pay-as-You-Go or Promo
Zones	1,000 zones	1,000 zones
Records	25,000 per zone	25,000 per zone

Resources	Monthly Universal Credits	Pay-as-You-Go or Promo
Zone File Size	1 MB	1 MB

Email Delivery Limits

Limits apply to each tenant or availability domain, as specified. There are three availability domains per region.

Resource	Monthly Universal Credits	Pay-As-You-Go or Promo
Email volume	50,000 emails per day	200 emails per day
Maximum approved senders	10,000	2,000
SMTP credentials	2 per user	2 per user
Sending rate	18,000 emails per minute	10 emails per minute



Note

Message size is limited to 2 MB, inclusive of message headers, body, and attachments.

File Storage Limits

Limits apply to each tenant or availability domain, as specified. There are three availability domains per region.

Resource	Pre-Paid	Pay-As-You-Go
File systems	100 per tenant per availability domain	100 per tenant per availability domain
Mount targets	2 per tenant per availability domain	2 per tenant per availability domain
Maximum file system size	8 exabytes	8 exabytes

Health Checks Limits

Health Checks limits are **global**.

Resource	Monthly Universal Credits	Pay-as-You-Go or Promo
Endpoint tests	1000 per account	1000 per account

IAM Limits

IAM limits are **global**.

Resource	Monthly Universal Credits	Pay-as-You-Go or Promo
Users in a tenancy	250	250
Groups in a tenancy	250	250
Compartments in a tenancy	50	50
Policies in a tenancy	100	100
Statements in a policy	50	50
Users per group in a tenancy	100	100
Groups per user in a tenancy	50	50
Identity providers in a tenancy	3	3
Group mappings for an identity provider	50	50

Key Management Limits

Key Management limits are **global**.

Resources	Monthly Universal Credits	Pay-as-You-Go or Promo
Vaults in a tenancy	Contact Us	Contact Us
Keys in a vault (Key versions, whether enabled or disabled, count against your limits.)	Contact Us	Contact Us

The Key Management service is not available to promo customers.

Load Balancing Limits

Load Balancing limits are **regional**.

Resource	Monthly Universal Credits	Pay-as-You-Go or Promo
LB-Capacity-100Mbps	3 Load Balancers	1 Load Balancer
LB-Capacity-400Mbps	3 Load Balancers	1 Load Balancer
LB-Capacity-8000Mbps	Contact Us	Contact Us

Monitoring Limits

Monitoring limits are **regional**.

Resource	Monthly Universal Credits	Pay-as-You-Go or Promo
Alarms	50	50
Metrics (posted by services)	Unlimited	Unlimited

Networking Limits

Networking service limits apply to different scopes, depending on the resource.

VCN and Subnet Limits

Resource	Scope	Monthly Universal Credits	Pay-as-You-Go or Promo
VCN	Region	10	10
Subnets	VCN	300	300

Gateway Limits

Resource	Scope	Monthly Universal Credits	Pay-as-You-Go or Promo
Dynamic routing gateways (DRGs)	Region	5	5
Internet gateways	VCN	1*	1*
Service gateways	VCN	1	1
NAT gateways	VCN	1	1
* Limit for this resource cannot be increased			

IP Address Limits

Resource	Scope	Monthly Universal Credits	Pay-as-You-Go or Promo
Reserved public IPs	Region	50	50
Ephemeral public IPs	Instance	2 per VM instance 16 per bare metal instance	2 per VM instance 16 per bare metal instance

DHCP Option Limits

Resource	Scope	Monthly Universal Credits	Pay-as-You-Go or Promo
DHCP options	VCN	300	300

Route Table Limits

Resource	Scope	Monthly Universal Credits	Pay-as-You-Go or Promo
Route tables	VCN	300	300
Route rules	Route table	100*	100*
* Limit for this resource cannot be increased			

Security List Limits

Resource	Scope	Monthly Universal Credits	Pay-as-You-Go or Promo
Security lists	VCN	300	300
Security lists	Subnet	5*	5*
Ingress and egress rules	Security list	200 ingress rules* and 200 egress rules*	200 ingress rules* and 200 egress rules*
* Limit for this resource cannot be increased			

IPSec VPN Connection Limits

Resource	Scope	Monthly Universal Credits	Pay-as-You-Go or Promo
IPSec VPN connections	Region	4	4
Customer-premises equipment objects (CPEs)	Region	10	10
Dynamic routing gateways (DRGs)	Region	See Gateway Limits	See Gateway Limits

FastConnect Limits

Resource	Scope	Monthly Universal Credits	Pay-as-You-Go or Promo
Cross-connects	Region	Contact Us	Contact Us
Virtual circuits	Region	10	10
Dynamic routing gateways (DRGs)	Region	See Gateway Limits	See Gateway Limits

Notifications Limits

Notifications limits are **regional**.

Resource	Monthly Universal Credits	Pay-as-You-Go or Promo
Topics	50 (Active or Creating*) per tenancy	Contact Us
Subscriptions	10 (Active or Pending*) per topic 100 (Pending*) per tenancy	Contact Us

*** A lifecycle state. See NotificationTopic Reference and Subscription Reference.**

Object Storage and Archive Storage Limits

Object Storage and Archive Storage limits are **regional**.

Resource	Monthly Universal Credits	Pay-as-You-Go or Promo
Buckets	1000 per compartment	1000 per compartment
Objects per bucket	Unlimited	Unlimited
Maximum object size	10 TiB	10 TiB
Maximum object part size	50 GiB	50 GiB

Registry Limits

Registry limits are **regional**.

Resource	Monthly Universal Credits	Pay-as-You-Go or Promo
Repositories	500 repositories per OCI region	500 repositories per OCI region
Images	500 images per repository	500 images per repository

Resource Manager Limits

Resource Manager limits are **regional**.

Resource	Monthly Universal Credits	Pay-as-You-Go or Promo
Stacks	100 stacks per tenant	10 stacks per tenant
Variables per stack	100	100

Resource	Monthly Universal Credits	Pay-as-You-Go or Promo
Zip file per stack	11 MB	11 MB
Jobs (concurrent)	5 per tenant	1 per tenant
Job Duration	24 hours	24 hours

Traffic Management Steering Policies Limits

Traffic Management Steering Policies limits are **global**.

Resource	Monthly Universal Credits	Pay-as-You-Go or Promo
Policies	100 per tenant	100 per tenant
Attachments	1,000 per tenant	1,000 per tenant

WAF Limits

WAF limits are **global**.

Resource	Monthly Universal Credits	Pay-as-You-Go or Promo
Policies	50 per tenant	50 per tenant

Console Announcements

This topic describes the announcements that Oracle Cloud Infrastructure displays in the Console. Console announcements appear at the top of the page to communicate timely, important information about service status. You can also view a list of past announcements.

Types of Announcements

There are different categories of announcements. An announcement's prefix helps you understand, at a glance, the type and relative severity of the information and whether there's anything you can or must do. Announcement types currently include the following, in order of most important to least:

- **Required action.** You must take specific action within your environment.
- **Emergency change.** There is a time period during which an unplanned, but urgent, change associated with your environment will take place.
- **Recommended action.** You have specific action to take within your environment, but the action is not required.
- **Planned change.** There is a time period during which a planned change associated with your environment will take place.
- **Planned change extended.** The scheduled change period has extended beyond what was previously communicated.
- **Planned change rescheduled.** The planned change to your environment has been postponed to a later time or date.
- **Production event.** An impactful change to your environment either recently occurred or is actively occurring.
- **Planned change completed.** The planned change to your environment has been completed and regular operations have resumed.
- **Information.** There is information that you might find useful, but is not urgent and does not require action on your part.

For announcements that require action and affect Oracle Cloud Infrastructure Compute instances, you will get 30 days of advance notice. If you need to delay the actions described in the announcement, contact support to request one of the alternate dates listed in the announcement. Critical vulnerabilities might not be eligible for delay.

Authentication and Authorization

Depending on whether you have access, you might not see any announcements. With access to announcements, you can either see only the summary version of any given announcement or you can also view announcement details.

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

Email Delivery

As part of your service agreement, Oracle Cloud Infrastructure also contacts you about service status announcements through email. These emails help alert you to upcoming changes that will impact your tenancy, such as those involving data centers or instances you use, or about required action on your part. Oracle sends these announcements to the tenant administrator email address and you cannot opt out of receiving this operational information. Whenever possible, we try to provide advance notice of impactful events. If you want to change the tenant administrator email address, contact Oracle Support. For more information, see Contacting Support.

Viewing Announcements

This section describes how to view announcements. The Console displays announcements as banners that span the width of the top of your browser window. As long as an announcement remains in effect and you have the access to view announcements, the banner announcement displays each time you sign in to the Console until you mark it as read. You can also view all past announcements. The Announcements icon displays a green dot if you have any unread announcements.

To dismiss a banner announcement

- To close a banner announcement until the next time you sign in to the Console, click the X at the far right edge of the banner. If you want to stop seeing an announcement as a banner altogether, you must mark it as read. For more information, see To mark an announcement as read.

To view the details of an announcement

1. Do one of the following:
    - If you are viewing a banner, click the **Show details** link near the far right edge of the banner.
    - If you are viewing a list of announcements, under the **Summary** column, click the announcement summary.
2. On the **Announcement Details** page, you can view the following information:
    - **Description.** This describes the issue or event in greater detail than the summary text of the announcement.
    - **OCID.** This is the announcement's unique, Oracle-assigned identifier.
    - **Reference Ticket Number.** You can use this number to refer to the issue when talking to Support.

- **Type.** This is one of several predefined categories that helps to set expectations about the nature and severity of the issue described.
- **Affected Service.** This indicates the Oracle Cloud Infrastructure services affected by the issue or event.
- **Region.** This tells you what Oracle Cloud Infrastructure regions are impacted.
- **Start Time.** This is when the issue or event was first detected.
- **End Time.** This is when the issue or event was resolved.
- **Required After.** This is the date after which you must address any required actions described in the announcement.
- **Created.** This is when the announcement was created.
- **Updated.** This is when the announcement was updated.
- **Additional Information.** This includes information such as workarounds or background material.
- **Impacted Resources.** This shows the resources that were affected in some way by the event that prompted the announcement.

3. Optionally, if you want to refer to the list of impacted resources later, click **Download Impacted Resources List**.

To view a list of all announcements

1. Click the Announcements icon (🔔).
2. The **Announcements** page displays all announcements. From this page, you can do the following:
   - **Filter.** You can filter announcements by type or by start or end date.
   - **Sort.** You can sort announcements by summary, type, event start time, or publish time (which indicates when the announcement was last updated).
   - **Mark as read.** You can mark announcements as read if you want stop seeing

them as banners in the Console in subsequent sessions.

- **View announcement details.** You can view the details of an announcement.

To filter a list of announcements

1. Click the Announcements icon (🔔).
2. To filter the list, under **Filters**, do one of the following:
   - Click **Type**, and then click a type from the list.
   - Click **Start Date**, and then choose a date to see only events that started on that date.
   - Click **End Date**, and then choose a date to see only events that ended on that date.
3. To clear a filter on a date, click the X next to the date.

To sort a list of announcements

1. Click the Announcements icon (🔔).
2. By default, the list displays announcements according to the event start time, from most recent to least. To sort the list another way, do one of the following:
   - Click **Summary**. The list sorts alphabetically, according to the summary of the announcement.
   - Click **Type**. The list sorts according to the importance of the announcement.
   - Click **Start Time**. The list sorts according to the start time of the event described in the announcement. If you begin by viewing the default sort order, the sort order will change to show the oldest announcement at the beginning of the list.
   - Click **Publish Time**. The list sorts according to the time that an announcement was last updated. You might find it helpful to sort by this column if you want to track an ongoing issue or if an announcement requires action on your part.
3. To sort the list again, repeat the previous step.

To mark an announcement as read

1. Click the Announcements icon (🔔).
2. Find the announcement that you want to mark as read, click the Actions icon (three dots), and then click **Mark As Read**.

Using the Command Line Interface (CLI)

For information about using the CLI, see Command Line Interface (CLI). For a complete list of flags and options available for CLI commands, see CLI Help.

To view the details of an announcement

Open a command prompt and run `oci announce announcements get` to view detailed information about an announcement:

```
oci announce announcements get --announcement-id <announcement_OCID>
```

For example:

```
oci announce announcements get --announcement-id
ocid1.announcement.region1..examplear73oue4jdywjjvietoc6im3cvb6xae4falm3faux5us3iwra3t6q
```

To view a list of all announcements

Open a command prompt and run `oci announce announcements list` to view a list of all announcements:

```
oci announce announcements list --compartment-id <compartment_OCID>
```

For example:

```
oci announce announcements list --compartment-id
ocid1.tenancy.oc1..exampleati4wjo6cvbxq4iusld5ltpneskcfy7lr4a6wfauxuwrwed5bsdea
```

To filter a list of announcements

Open a command prompt and run `oci announce announcements list` to filter a list of announcements.

To filter a list of announcements by announcement type:

```
oci announce announcements list --compartment-id <compartment_OCID> --announcement-type <announcement_type>
```

For example:

```
oci announce announcements list --compartment-id
ocid1.tenancy.oc1..exampleati4wjo6cvbxq4iusld5ltpneskcfy7lr4a6wfauxuwrwed5bsdea --announcement-type
ACTION_REQUIRED
```

To sort a list of announcements

Open a command prompt and run `oci announce announcements list` to sort a list of announcements.

To sort a list of announcements in ascending order of time created, from oldest to newest:

```
oci announce announcements list --compartment-id <compartment_OCID> --sort-order ASC
```

For example:

```
oci announce announcements list --compartment-id
ocid1.tenancy.oc1..exampleati4wjo6cvbxq4iusld5ltpneskcfy7lr4a6wfauxuwrwed5bsdea --sort-order ASC
```

To mark an announcement as read

Open a command prompt and run `oci announce user-status update` to mark an announcement as read:

```
oci announce user-status update --announcement-id <announcement_OCID> --user-status-announcement-id
<announcement_OCID> --user-id <user_OCID> --time-acknowledged <date_and_time>
```

For example:

```
oci announce user-status update --announcement-id
ocid1.announcement.region1..examplear73oue4jdywjjvietoc6im3cvb6xae4falm3faux5us3iwra3t6q --user-status-
announcement-id ocid1.announcement.region1..examplear73oue4jdywjjvietoc6im3cvb6xae4falm3faux5us3iwra3t6q
--user-id ocid1.user.region1..exampleaorxz3psplonigcvbzy5oaiwiubh7k7ip6zgklfauxic67kksu4oq --time-
acknowledged 2019-01-06T20:14:00+00:00
```

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operations to manage announcements:

- GetAnnouncement
- GetAnnouncementUserStatus
- ListAnnouncements
- UpdateAnnouncementUserStatus

Prerequisites for Oracle Platform Services on Oracle Cloud Infrastructure

This topic describes procedures that are required by some Oracle Platform Services before you can launch them on Oracle Cloud Infrastructure. The information in this topic applies only to the following services:

- Oracle Big Data Cloud
- Oracle Database Cloud Service
- Oracle Data Hub Cloud Service
- Oracle Event Hub Cloud Service
- Oracle Java Cloud Service
- Oracle MySQL Cloud Service
- Oracle SOA Cloud Service

For a list of all services supported on Oracle Cloud Infrastructure, see Information About Supported Platform Services.

Accessing Oracle Cloud Infrastructure

Oracle Cloud Infrastructure has a different interface and credential set than your Oracle Platform Services. You can access Oracle Cloud Infrastructure using the Console (a browser-based interface) or the REST API. Instructions for the Console and API are included in topics throughout this guide. For a list of available SDKs, see Software Development Kits and Command Line Interface.

To access the Console, you must use a supported browser (Oracle Cloud Infrastructure supports the latest desktop versions of Google Chrome, Microsoft Edge, Internet Explorer 11, Safari, Firefox, and Firefox ESR. Note that private browsing mode is not supported for Firefox, Internet Explorer, or Edge. Mobile browsers are not supported.

Required Identity and Access Management (IAM) Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

See Common Policies for more information and examples.

Resources Created in Your Tenancy by Oracle

Oracle creates a compartment in your tenancy for Oracle Platform Services. This compartment is specially configured by Oracle for the Oracle Cloud Infrastructure resources that you create through the Platform Services. You can't choose another compartment for Oracle to use.

Along with this compartment, Oracle creates the IAM policies to allow Oracle Platform Services access to the resources.

The compartment that Oracle creates for Oracle Platform Services is named: `ManagedCompartmentForPaaS`.

The polices that Oracle creates for Oracle Platform Services are:

- `PSM-root-policy`

  This policy is attached to the root compartment of your tenancy.
- `PSM-mgd-comp-policy`

  This policy is attached to the `ManagedCompartmentForPaaS` compartment.



Warning

Do not make any changes to these resources. Editing or renaming the policies or the compartment can result in loss of functionality.

Prerequisites for Oracle Platform Services

Before you can create instances of an Oracle Platform Service on Oracle Cloud Infrastructure, you need to have the following resources in your Oracle Cloud Infrastructure tenancy:

- A compartment for your resources
- A virtual cloud network (VCN) with at least one public subnet
- IAM policies to allow Oracle Platform Services to access the VCN
- An Object Storage bucket
- Credentials to use with Object Storage

Some of the Platform Services automatically create some of these resources for you. See details about your service in the following sections.

Setting Up the Prerequisites



Note

To use **Autonomous Data Warehouse Cloud**, you don't need to set up any of the resources listed in this prerequisites section. However, if you optionally choose to use Oracle Cloud Infrastructure Object Storage for data loading, you need to perform these two tasks:

Create a bucket

Create an auth token

Following are two scenarios with procedure sets. If you need to set up all the required resources, follow Scenario 1. If you already have a VCN in your Oracle Cloud Infrastructure tenancy that you want to use for Oracle Platform Services, follow Scenario 2.

To follow a tutorial on how to set up the prerequisites for Scenario 1, see Creating the Infrastructure Resources Required for Oracle Platform Services.

Scenario 1: I need to create all the prerequisite resources

Create a compartment



Important

You cannot use the `ManagedCompartmentForPaaS` for your VCN and bucket.

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Compartments**.
2. A list of the existing compartments in your tenancy is displayed.

3. Click **Create Compartment**.
4. Enter the following:
    - **Name:** For example, PaaSResources. Restrictions for compartment names are: Maximum 100 characters, including letters, numbers, periods, hyphens, and underscores. The name must be unique across all the compartments in your tenancy
    - **Description:** A friendly description.
5. Click **Create Compartment**.

Set up your virtual cloud network

This procedure creates a VCN with these characteristics:

- A VCN with CIDR 10.0.0.0/16.
- Three public subnets (10.0.0.0/24, 10.0.1.0/24, and 10.0.2.0/24) each using the VCN's default security list, default route table, and default DHCP options.
- An internet gateway, with the required route rule in the default route table.
- Use of the Internet and VCN Resolver for DNS, so your instances can use their hostnames instead of their private IP addresses to communicate with each other.



Tip

This Quick VCN procedure is useful for getting started and trying out Oracle Platform Services on Oracle Cloud Infrastructure. For production, use the procedure in VCNs and Subnets. That topic explains features such as how to specify the CIDR ranges for your VCN and subnets, and how to secure your network. When you use the advanced procedure, remember that the VCN that you create must have a public subnet for Oracle Platform Services to use.

1. Open the **Region** menu and select the region in which you want to create the Oracle PaaS service instance.
   Select a region that's within the default data region of your account. For example, if your default data region is EMEA, then select eu-frankfurt-1 or uk-london-1.
2. From the **Compartment** list, select the compartment you created.
3. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
4. Click **Create Virtual Cloud Network**.
5. In **Create in Compartment**, leave the default value (the compartment you're currently working in).
6. Enter a friendly name for the cloud network, for example: `PaaSVCN`. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API).
7. Select **Create Virtual Cloud Network Plus Related Resources**.
8. Scroll down to the bottom of the dialog box and click **Create Virtual Cloud Network**.

Permit Oracle Platform Services to access resources

1. In the Console, navigate to the root compartment of your tenancy by clicking your tenancy name in the **Compartment** list.
2. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Policies**.
3. Click **Create Policy**.
4. Enter the following:
    - **Name:** A unique name for the policy. The name must be unique across all policies in your tenancy. You cannot change this later.
    - **Description:** A friendly description. You can change this later if you want to.
    - **Policy Versioning:** Select **Keep Policy Current** if you'd like the policy to stay current with any future changes to the service's definitions of verbs and resources. Or if you'd prefer to limit access according to the definitions that were current on a specific date, select **Use Version Date** and enter that date in format YYYY-MM-DD format. For more information, see Policy Language Version.
    - **Statement:** To allow Oracle Platform Services access to use the network in your compartment, enter the following policy statements. Replace *<compartment_name>* with your compartment name. Click **+** after each statement to add another.

      ```
      Allow service PSM  to inspect vcns in compartment <compartment_name>
      ```

      ```
      Allow service PSM  to use subnets in compartment <compartment_name>
      ```

      ```
      Allow service PSM  to use vnics in compartment <compartment_name>
      ```

      ```
      Allow service PSM  to manage security-lists in compartment <compartment_name>
      ```

      For more information about policies, see Policy Basics and also Policy Syntax.
5. (Optional) If you want to enable the use of an Autonomous Transaction Processing or Oracle Cloud Infrastructure Database instance in your compartment as the infrastructure schema database for your Oracle Java Cloud Service instance, then add the following statements:

```
Allow service PSM to inspect autonomous-database in compartment <compartment_name>

Allow service PSM to inspect database-family in compartment <compartment_name>
```

6. Click **Create**.

Create a bucket

1. Open the **Region** menu and select the region in which you want to create the Oracle PaaS service instance.
   Select a region that's within the default data region of your account. For example, if your default data region is EMEA, then select eu-frankfurt-1 or uk-london-1.
2. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
3. Choose the compartment you created.
4. Click **Create Bucket**.
5. In the **Create Bucket** dialog, enter a bucket name, for example: `PaasBucket`.
   Make a note of the name you enter. You will need it when you create an instance for your Oracle Platform Service later.
6. Click **Create Bucket**.

Set up credentials to use with Object Storage

For Big Data Cloud, set up an API signing key:

Set up an API signing key

Follow the instructions in this topic: Required Keys and OCIDs.

For all other services, create an auth token. Note that your service might refer to this credential as a Swift password. Use the auth token wherever you are asked to provide a Swift password.

Create an auth token

1. View the user's details:
   - If you're creating an auth token for yourself: Open the **User** menu () and click **User Settings**.
   - If you're an administrator creating an auth token for another user: In the Console, click **Identity**, and then click **Users**. Locate the user in the list, and then click the user's name to view the details.
2. On the left side of the page, click **Auth tokens**.
3. Click **Generate Token**.
4. Enter a friendly description for the token and click **Generate Token**.
   The new token is displayed.
5. Copy the token immediately, because you can't retrieve it again after closing the dialog box. Also, make sure you have this token available when you create your Oracle Platform Services instance.

Scenario 2: I have an existing VCN in Oracle Cloud Infrastructure that I want to use for my Oracle Platform Services instance

You can use an existing VCN. The VCN must have at least one public subnet. Perform these tasks to complete the prerequisites:

Permit Oracle Platform Services to access resources

1. In the Console, navigate to the root compartment of your tenancy by clicking your tenancy name in the **Compartment** list.
2. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Policies**.
3. Click **Create Policy**.

4. Enter the following:
    - **Name:** A unique name for the policy. The name must be unique across all policies in your tenancy. You cannot change this later.
    - **Description:** A friendly description. You can change this later if you want to.
    - **Policy Versioning:** Select **Keep Policy Current** if you'd like the policy to stay current with any future changes to the service's definitions of verbs and resources. Or if you'd prefer to limit access according to the definitions that were current on a specific date, select **Use Version Date** and enter that date in YYYY-MM-DD format. For more information, see Policy Language Version.
    - **Statement:** To allow Oracle Platform Services access to use the network, enter the following policy. Click **+** after each statement to add another. In each statement, replace *<compartment_name>* with the name of the compartment where your VCN resides.

      ```
      Allow service PSM  to inspect vcns in compartment <compartment_name>
      ```

      ```
      Allow service PSM  to use subnets in compartment <compartment_name>
      ```

      ```
      Allow service PSM  to use vnics in compartment <compartment_name>
      ```

      ```
      Allow service PSM  to manage security-lists in compartment <compartment_name>
      ```

      For more information about policies, see Policy Basics and also Policy Syntax.
5. (Optional) If you want to enable the use of an Autonomous Transaction Processing or Oracle Cloud Infrastructure Database instance in your compartment as the infrastructure schema database for your Oracle Java Cloud Service instance, then add the following statements:

   ```
   Allow service PSM to inspect autonomous-database in compartment <compartment_name>
   ```

   ```
   Allow service PSM to inspect database-family in compartment <compartment_name>
   ```
6. Click **Create**.

Create a bucket

1. Open the **Region** menu and select the region in which you want to create the Oracle PaaS service instance.
   Select a region that's within the default data region of your account. For example, if your default data region is EMEA, then select eu-frankfurt-1 or uk-london-1.
2. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
3. Choose the compartment you want to create the bucket in.
4. Click **Create Bucket**.
5. In the **Create Bucket** dialog, enter a bucket name, for example: `PaasBucket`. Make a note of the name you enter. You will need it when you create an instance for your Oracle Platform Service later.
6. Click **Create Bucket**.

Set up credentials to use with Object Storage

For Big Data Cloud, set up an API signing key:

Set up an API signing key

Follow the instructions in this topic: Required Keys and OCIDs.

For all other services, create an auth token. Note that your service might refer to this credential as a Swift password. Use the auth token wherever you are asked to provide a Swift password.

Create an auth token

1. View the user's details:

- If you're creating an auth token for yourself: Open the **User** menu () and click **User Settings**.
- If you're an administrator creating an auth token for another user: In the Console, click **Identity**, and then click **Users**. Locate the user in the list, and then click the user's name to view the details.

2. On the left side of the page, click **Auth Tokens**.
3. Click **Generate Token**.
4. Enter a friendly description for the token and click **Generate Token**.
   The new token is displayed.
5. Copy the auth token immediately, because you can't retrieve it again after closing the dialog box. Also, make sure you have this token available when you create your Oracle Platform Services instance.

Information About Supported Platform Services

The following table lists the services supported on Oracle Cloud Infrastructure and links to more information about using those services on Oracle Cloud Infrastructure:

Service	More Information
Autonomous Analytics Cloud	About Oracle Autonomous Analytics Cloud
Autonomous API Platform Cloud Service	Using Oracle Autonomous API Platform Cloud Service
Autonomous Data Warehouse Cloud	About Autonomous Data Warehouse Cloud
Integration Cloud	Oracle Integration Cloud

Service	More Information
Autonomous Mobile Cloud Enterprise	About Oracle Autonomous Mobile Cloud Enterprise
NoSQL Database Cloud	Oracle NoSQL Database Cloud
Oracle Visual Builder	Administering Oracle Visual Builder
Big Data Cloud	About Big Data Cloud Clusters in Oracle Cloud Infrastructure
Content and Experience Cloud	Administering Oracle Content and Experience Cloud
Data Hub Cloud Service	About Oracle Data Hub Cloud Service Clusters in Oracle Cloud Infrastructure
Data Integration Platform Cloud	What is Oracle Data Integration Platform Cloud
Database Cloud Service	About Database Deployments in Oracle Cloud Infrastructure
Developer Cloud Service	About Oracle Developer Cloud Service in Oracle Cloud Infrastructure
Event Hub Cloud Service	About Oracle Event Hub Cloud Service - Dedicated Instances in Oracle Cloud Infrastructure
Java Cloud Service	About Java Cloud Service Instances in Oracle Cloud Infrastructure
MySQL Cloud Service	About MySQL Cloud Service Deployments in Oracle Cloud Infrastructure
Oracle SOA Cloud Service	About SOA Cloud Service Instances in Oracle Cloud Infrastructure Classic and Oracle Cloud Infrastructure

Billing and Payment Tools Overview

Oracle Cloud Infrastructure provides various billing and payment tools that make it easy to manage your service costs.

Budgets

Budgets can be used to set thresholds for your Oracle Cloud Infrastructure spending. You can set alerts on your budget to let you know when you might exceed your budget, and you can view all of your budgets and spending from one single place in the Oracle Cloud Infrastructure console.

See Budgets Overview for more information.

Cost Analysis

Cost Analysis provides easy-to-use visualization tools to help you track and optimize your Oracle Cloud Infrastructure spending. For more information, see Checking Your Balance and Usage.

Usage Reports

A usage report is a comma-separated value (CSV) file that can be used to get a detailed breakdown of resources in Oracle Cloud Infrastructure for audit or invoice reconciliation.

For more information, see Usage Reports Overview.

Payment Methods

The Payment Method section of the Oracle Cloud Infrastructure Console allows you to easily manage how you pay for your Oracle Cloud Infrastructure usage.

For more information, see Changing Your Payment Method.

Built 5/24/2019 8:10 AM

Budgets Overview

A budget can be used to set soft limits on your Oracle Cloud Infrastructure spending. You can set alerts on your budget to let you know when you might exceed your budget, and you can view all of your budgets and spending from one single place in the Oracle Cloud Infrastructure console.

How Budgets Work

Budgets are set on compartments (including the root compartment) to track all spending in that compartment and its children, so it's important to structure resources into compartments if you want to create granular budgets.

All budgets alerts are evaluated every 15 minutes. To see the last time a budget was evaluated, open the details for a budget. You will see fields that show the current spend, the forecast and the "Spent in period" field which shows you the time period over which the budget was evaluated. When a budget alert fires, the email recipients configured in the budget alert receive an email.

Budget Concepts

The following concepts are essential to working with budgets:

Budget

A monthly threshold you define for your Oracle Cloud Infrastructure spending. Budgets are set on compartments and track all spending in the compartment and any child compartments. Note: the budget tracks spending in the specified target compartment, but you need to have permissions to manage budgets in the root compartment of the tenancy to create and use budgets.

Alert

You can define email alerts that get sent out for your budget. You can send a customized email message body with these alerts. Alerts are evaluated every 15 minutes, and can be triggered when your actual or your forecasted spending hits either a percentage of your budget or a specified set amount.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

If you're new to policies, see Getting Started with Policies and Common Policies.

To use budgets, you must be in a group that can use "usage-budgets" in the tenancy (which is the root compartment) or be able to use all resources in the tenancy. All budgets are created in the root compartment, regardless of the compartment they are targeting, so IAM policies that grant budget permissions outside of the root will not be meaningful.

IAM Policy	Description
Allow group accountants to inspect usage-budgets in tenancy	Accountants can inspect budgets including spend.
Allow group accountants to read usage-budgets in tenancy	Accountants can read budgets including spend (same as list).
Allow group accountants to use usage-budgets in tenancy	Accountants can create and edit budgets and alerts rules.
Allow group accountants to manage usage-budgets in tenancy	Accountants can create, edit, and delete budgets and alerts rules.

Tagging Resources

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

Managing Budgets

This topic discusses how to view and manage your budgets.

USING THE CONSOLE

To create a budget

1. Open the navigation menu. Under **Governance and Administration**, go to **Billing** and click **Budgets**.
2. Click **Create Budget** at the top of the budgets list.
3. In the **Create Budget** dialog, configure your budget:
    a. Select a target compartment for your budget from the **Target Compartment** drop-down list. Note that while the budget tracks spending in the specified target compartment, but you need to have permissions to manage budgets in the root compartment of the tenancy to create and use budgets.

b. Enter a name for your budget in the **Name** text field. The name can only contain alphanumeric characters, dashes, and the underscore character, and can't begin with a number.

c. Enter a monthly amount for your budget in the **Monthly Budget Amount** field. The minimum allowed value for your monthly budget is 1; the maximum allowed value is 999,999,999,999.

4. You can optionally create an alert for your budget by creating a budget alert rule. In the Budget Alert Rule panel on the Create Budget dialog, configure your alert rule:

   a. Select a threshold for your alert from the **Threshold Metric** drop-down list. There are two possible values:

      Actual Spend will watch the actual amount you spend in your compartment per month;

      Forecast Spend will watch your resource usage and alert you when it appears that you'll exceed your budget. The forecast algorithm is linear extrapolation and requires at least 3 days of consumption to trigger

   b. Select a threshold type from the **Threshold Type** drop-down list. You can select either a percentage of your monthly budget (which must be greater than 0 and no greater than 10,000) or a fixed amount.

   c. The label of the next text field changes depending on what type of threshold you selected. Enter either a **Threshold %** or a **Threshold Amount**.

   d. In the **Email Recipients** field, enter one or more email addresses to receive the alerts. Multiple addresses can be separated using a comma, semicolon, space, tab, or new line.

   e. Enter the body of your email alert in the **Email Message** field. The text of the email message cannot exceed 1000 characters. This message will be included with metadata about your budget, including the budget name, the compartment, and the amount of your monthly budget. You can use this message to for things like providing instructions to the recipient that explain how to request a budget increase or reminding users about corporate policies.

5. **Advanced Options (optional):** Click the **Show Advanced Options** link to add **Tags** to your Budget. Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
6. Click the **Create** button to create your budget.

To view or edit a budget

1. Open the navigation menu. Under Governance and Administration, go to Billing and click **Budgets**.
2. From the list of budgets, click on the budget you want to edit. The budget detail screen will appear.
3. Click the **Edit** button. The **Edit Budget** dialog will appear.
4. You can edit the name of your budget or the budget amount.
5. When you are finished, click **Save Changes**.

To delete a budget

1. From the list of budgets, select **Delete** from the context menu, or click the **Delete** button at the top of budget detail screen. The **Confirm Delete** dialog will appear.
2. Click the **Confirm** button to delete the budget, or cancel by clicking **Cancel**.

To manage tags for a budget

1. Open the navigation menu. Under Governance and Administration, go to Billing and click **Budgets**.
2. From the list of budgets, click on the budget you want to tag. The budget detail screen will appear.

3. Click the **Add tag(s)** button to add a tag.
4. Click the the **Tags** tab and then click on the pencil icon next to a tag you want to edit or remove.

USING THE API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operation to manage budgets:

- ListBudgets
- GetBudget
- CreateBudget
- DeleteBudget
- UpdateBudget

Managing Budget Alert Rules

You can set email alerts on your budgets. You can set alerts that are based on a percentage of your budget or an absolute amount, and on your actual spending or your forecast spending.

This topic covers how to view and manage your budget alert rules.

REQUIRED IAM POLICY

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

If you're new to policies, see Getting Started with Policies and Common Policies.

USING THE CONSOLE

To create a budget alert rule

1. Click the budget that you want to create an alert for from the budgets list.
2. In the **Budget Alert Rules** panel at the bottom of the screen, click the **Create Budget Alert Rule** button.
3. Configure your alert rule:
    a. Select a threshold for your alert from the **Threshold Metric** drop-down list. There are two possible values:

    Actual Spend will watch the actual amount you spend in your compartment per month;

    Forecast Spend will watch your resource usage and alert you when it appears that you'll exceed your budget. The forecast algorithm is linear extrapolation and requires at least 3 days of consumption to trigger

    b. Select a threshold type from the **Threshold Type** drop-down list. You can select either a percentage of your monthly budget (which must be greater than 0 and no greater than 10,000) or a fixed amount.

    c. The label of the next text field changes depending on what type of threshold you selected. Enter either a **Threshold %** or a **Threshold Amount**.

    d. In the **Email Recipients** field, enter one or more email addresses to receive the alerts. Multiple addresses can be separated using a comma, semicolon, space, tab, or new line.

    e. Enter the body of your email alert in the **Email Message** field. The text of the email message cannot exceed 1000 characters. This message will be included with metadata about your budget, including the budget name, the compartment, and the amount of your monthly budget. You can use this message to for things like providing instructions to the recipient that explain how to request a budget increase or reminding users about corporate policies.
4. Click the **Create** button to create your alert.

To view or edit a budget alert rule

1. In the list of budget alert rules, click the menu icon at the right side of the list and select **View/Edit** from the context menu.
2. Edit your alert rule.
3. Confirm your changes by clicking **Save Changes**, or dismiss the dialog without saving by clicking the **Cancel** button.

To delete a budget alert rule

1. In the list of budget alert rules, click the menu icon at the right side of the list and select **Delete** from the context menu.
2. Confirm or cancel the delete operation in the **Confirm Delete** dialog by clicking either the **Confirm** or **Cancel** button.

USING THE API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operation to manage budget alert rules:

- ListAlertRules
- GetAlertRule
- CreateAlertRule
- DeleteAlertRule
- UpdateAlertRule

Usage Reports Overview

A usage report is a comma-separated value (CSV) file that can be used to get a detailed breakdown of your Oracle Cloud Infrastructure resources for audit or invoice reconciliation.

How Usage Reports Work

The usage report is automatically generated daily, and is stored in an Oracle-owned object storage bucket. It contains one row per each Oracle Cloud Infrastructure resource (such as instance, object storage bucket, VNIC) per hour along with consumption information, metadata, and tags. Usage reports generally contain 24 hours of usage data, although occasionally a usage report may contain late-arriving data that is older than 24 hours.

Usage reports are retained for one year.

The file name for each usage report is appended with an automatically incrementing numerical value.

Usage Report Schema

The following table shows the usage report schema.

Field Name	Description
`lineItem/referenceNo`	Line identifier. Used for debugging and corrections.
`lineItem/TenantId`	The identifier (OCID) for the Oracle Cloud Infrastructure tenant.
`lineItem/intervalUsageStart`	The start time of the usage interval for the resource in UTC.
`lineItem/intervalUsageEnd`	The end time of the usage interval for the resource in UTC.
`product/service`	The service that the resource is in.
`product/resource`	The resource name used by the metering system.
`product/compartmentId`	The ID of the compartment that contains the resource.

Field Name	Description
product/compartmentName	The name of the compartment that contains the resource.
product/region	The region that contains the resource.
product/availabilityDomain	The availability domain that contains the resource.
product/resourceId	The identifier for the resource.
usage/consumedQuantity	The quantity of the resource that has been consumed over the usage interval.
usage/billedQuantity	The quantity of the resource that has been billed over the usage interval.
usage/consumedQuantityUnits	The unit for the consumed quantity and billed quantity.
usage/consumedQuantityMeasure	The measure for the consumed quantity and billed quantity.
lineItem/isCorrection	For future use. Used if the current line is a correction.
tags/	The usage report contains one column per tag definition (includes all tag definitions, not just cost tracking tags).

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have

permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

If you're new to policies, see Getting Started with Policies and Common Policies.

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

Accessing Usage Reports

A usage report is a comma-separate value (CSV) file that is generated daily and stored in an object storage bucket. This topic describes how to access usage reports.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

If you're new to policies, see Getting Started with Policies and Common Policies.

Reports are generated in another tenancy and stored in an Oracle-owned object storage bucket. You must set up a cross-tenancy IAM policy to access your usage reports as shown below, changing the group name as appropriate:

```
define tenancy usage-report as
ocid1.tenancy.oc1..aaaaaaaaned4fkpkisbwjlr56u7cj63lf3wffbilvqknstgtvzub7vhqkggq
endorse group MyGroupName to read objects in tenancy usage-report
```

USING THE CONSOLE

To download a usage report

1. Open the navigation menu. Under **Governance and Administration**, go to **Billing** and select **Usage Report**.
2. Click the report you want to download from the list, and follow your brower's instructions for downloading.

USING THE API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

To download a usage report, use the Object Storage APIs. The reports are stored in the tenancy's home region. The object storage namespace used for the reports is `bling`; the bucket name is the tenancy OCID.

The following example shows how to download a usage report using a Python script:

```
import oci
import os

# This script downloads all of the usage reports for a tenancy (specified in the config file)
#
# Pre-requisites: Create an IAM policy to endorse users in your tenancy to read usage reports from the
OCI tenancy
#
# Example policy:
# define tenancy usage-report as
```

```
ocid1.tenancy.oc1..aaaaaaaaned4fkpkisbwjlr56u7cj63lf3wffbilvqknstgtvzub7vhqkggq
# endorse group group_name to read objects in tenancy usage-report
#
# Note - the only value you need to change is group name. Do not change the OCID in the first statement

       usage_report_namespace = 'bling'

       # Update these values
       destintation_path = 'downloaded_reports'

       # Make a directory to receive reports
       if not os.path.exists(destintation_path):
       os.mkdir(destintation_path)

       # Get the list of usage reports
       config = oci.config.from_file('config/config','DEFAULT')
       usage_report_bucket = config['tenancy']
       object_storage = oci.object_storage.ObjectStorageClient(config)
       report_bucket_objects = object_storage.list_objects(usage_report_namespace, usage_report_bucket)

       for o in report_bucket_objects.data.objects:
          print('Found file ' + o.name)
          object_details = object_storage.get_object(usage_report_namespace,usage_report_bucket,o.name)
          filename = o.name.rsplit('/', 1)[-1]

       with open(destintation_path + '/' + filename, 'wb') as f:
          for chunk in object_details.data.raw.stream(1024 * 1024, decode_content=False):
             f.write(chunk)

       print('Finished downloading ' + o.name + '\n')
```

Console Cookies and Local Storage

The Oracle Cloud Infrastructure console uses browser cookies and local storage as detailed below.

Oracle Cloud Infrastructure Console Login Page Cookies

Name	Purpose	Duration	Impact of Disabling
bmc_ tenancy	Tracks the default tenancy for the OCI Console login page	30 days	The last-used tenancy is not tracked, and users are asked to specify a tenancy when logging into the OCI Console

Oracle Cloud Infrastructure Console Local Storage

Name	Purpose	Duration	Impact of Disabling
hg-session-*<userid>* :*<session>*	UI State information, includes selected region, active compartment and other UI state	Never expires	Console UI may not work optimally
recorded-events	Temporary cache of console usage data. Does not include any sensitive information.	Never expires	Console usage not recorded for analysis and product improvement purposes
recorded-metrics	Temporary cache of console metrics (for example, page load time)	Never expires	Console metrics not recorded for analysis and product improvement purposes

Oracle Cloud Infrastructure Console Cookies

Name	Purpose	Duration	Impact of Disabling
s_fid	Adobe Analytics unique visitor information, anonymous	2 years	Cannot track users across different Oracle products (OCI, Cloud Marketplace)
s_nr	Adobe Analytics unique visitor information, anonymous	2 years	Cannot track users across different Oracle products (OCI, Cloud Marketplace)
gpw_e24	Records and saves the previously accessed URL (anonymous)	Never expires	Console metrics not recorded for analysis and product improvement purposes

Oracle Cloud Infrastructure Console Local Storage - Indexed DB

Table	Field	Purpose
duplo - *<tenancy ocid>*	activeCompartmentId	Stores the compartment a user has selected in the UI
duplo - *<tenancy ocid>*	activeRegionId	Stores the region a user has selected in the UI
duplo - *<tenancy ocid>*	selectedLocale	Stores the user's current locale

Table	Field	Purpose
opc-key-store	key	Signed ID token (carries user identity information) and a signed security token (carries transient user public key as a JSON Web Key) used for signed request calls to API end points. The security token expires after 24 hours, at which point the user will be prompted to log in again.
opc-key-store-v2	key	Signed ID token (carries user identity information) and a signed security token (carries transient user public key as a JSON Web Key) used for Signed Request calls to API end points. The security token expires after 24 hours, at which point the user will be prompted to log in again.

My Services Use Cases

Oracle's My Services dashboard is a place to check the overall status of your purchased services and manage your accounts or subscriptions, including Oracle Cloud Infrastructure. You can navigate directly to My Services from the Console: Open the navigation menu. Under **Solutions, Platform and Edge**, click **My Services Dashboard**.

To interact programmatically with My Services, you can use the Oracle Cloud My Services API. To help you get started, here are some use cases:

- Service Discovery Use Case
- Exadata Use Cases
- Managing Exadata Instances
- Using Access Token Authorization with My Services API

Service Discovery Use Case

This use case shows how you can get the list of your service entitlement IDs.

Discover Current Service Entitlement IDs

Many of the My Services API operations require you to specify the `serviceEntitlementId`. To get the list of all your service entitlement IDs, use the GET ServiceEntitlements operation. This operation returns information that you can use to make more specific requests using the Oracle Cloud My Services API.

Example:

```
GET /itas/<domain>/myservices/api/v1/serviceEntitlements
```



Note

In the examples, *<domain>* is the identity domain ID. An identity domain ID can be either the *IDCS GUID* that identifies the identity domain for the users within Identity Cloud Service (IDCS) or the *Identity Domain name* for a traditional Cloud Account.

To obtain the IDCS GUID

Go to the Users page in My Services dashboard and click **Identity Console**. The URL in the browser address field displays the IDCS GUID for your identity domain. For example:

```
https://idcs-
105bbbdfe5644611bf7ce04496073adf.identity.oraclecloud.com/ui/v
1/adminconsole/?root=users
```

In the above URL, `idcs-105bbbdfe5644611bf7ce04496073adf` is the IDCS GUID for your identity domain.

Example payload returned for this request:

```
{
     "items": [
         {
             "id": "cesi-511202718",                          // Unique ServiceEntitlementId
             "purchaseEntitlement": {                         // Purchase Entitlement is the entity
bought by a customer
                 "subscriptionId": "511203590",
                 "id": "511203590",
                 "canonicalLink": "/itas/<domain>/myservices/api/v1/purchaseEntitlements/511203590"
             },
             "serviceDefinition": {
                 "canonicalLink": "/itas/<domain>/myservices/api/v1/serviceDefinitions/500089778",
```

```
                "id": "500089778",
                "name": "Storage"                           // The customer is entitled to use the
Storage Service
            },
            "createdOn": "2017-12-20T16:23:23.326Z",
            "createdBy": "paul.smith@oracle.com",
            "modifiedOn": "2017-12-20T18:35:40.628Z",
            "modifiedBy": "paul.smith@oracle.com",
            "identityDomain": {                             // Identity Domain to which the Service
Entitlement is associated
                "id": "511203592",
                "name": "myenvironment",
                "displayName": "myenvironment"
            },
            "cloudAccount": {                               // Cloud Account to which the Service
Entitlement is associated
                "id": "cacct-be7475efc2c54995bc842d3379d35812",
                "name": "myenvironment",
                "canonicalLink": "/itas/<domain>/myservices/api/v1/cloudAccounts/cacct-
be7475efc2c54995bc842d3379d35812"
            },
            "status": "ACTIVE",                             // Current Status
            "serviceConfigurations": {                      // Specific configuration information such
as Exadata configuration
                "canonicalLink": "/itas/<domain>/myservices/api/v1/serviceEntitlements/cesi-
511202718/serviceConfigurations"
            },
            "canonicalLink": "/itas/{domain}/myservices/api/v1/serviceEntitlements/cesi-511202718"
        },
        {
            "id": "cesi-511202719",
            "purchaseEntitlement": {
                "subscriptionId": "511203590",
                "id": "511203590",
                "canonicalLink": "/itas/<domain>/myservices/api/v1/purchaseEntitlements/511203590"
            },
            "serviceDefinition": {
                "canonicalLink": "/itas/<domain>/myservices/api/v1/serviceDefinitions/500123193",
                "id": "500123193",
                "name": "Compute"                           // The customer is entitled to use the
Compute Service
            },
```

```
            "createdOn": "2017-12-20T16:23:23.326Z",
            "createdBy": "paul.smith@oracle.com",
            "modifiedOn": "2017-12-20T18:35:40.628Z",
            "modifiedBy": "paul.smith@oracle.com",
            "identityDomain": {
                "id": "511203592",
                "name": "myenvironment",
                "displayName": "myenvironment"
            },
            "cloudAccount": {
                "id": "cacct-be7475efc2c54995bc842d3379d35812",
                "name": "myenvironment",
                "canonicalLink": "/itas/<domain>/myservices/api/v1/cloudAccounts/cacct-
be7475efc2c54995bc842d3379d35812"
            },
            "status": "ACTIVE",
            "serviceConfigurations": {
                "canonicalLink": "/itas/<domain>/myservices/api/v1/serviceEntitlements/cesi-
511202719/serviceConfigurations"
            },
            "canonicalLink": "/itas/<domain>/myservices/api/v1/serviceEntitlements/cesi-511202719"
        },
        ...                                                 // More Service Entitlements could be
displayed
    ],
 "canonicalLink": "/itas/<domain>/myservices/api/v1/serviceEntitlements",
 "hasMore": false,
 "limit": 25,
 "offset": 0
 }
```

Exadata Use Cases

The following use case examples can get you started working with the Exadata operations available in the Oracle Cloud My Services API.



Important

These use cases are for use with Oracle Database Exadata Cloud Service. For more information, see Administering Oracle Database Exadata Cloud Service. These use cases **DO NOT** apply to Exadata DB Systems available in Oracle Cloud Infrastructure.

Exadata Firewall Whitelisting

To enable access to your Exadata Cloud Service instance, you can configure security rules and associate them with your instance. The security rules define a whitelist of allowed network access points.

The firewall provides a system of rules and groups. By default, the firewall denies network access to the Exadata Cloud Service instance. When you enable a security rule, you enable access to the Exadata Cloud Service instance. To enable access you must:

- Create a security group and create security rules that define specific network access allowances.
- Assign the security group to your Exadata Cloud Service instance.

You can define multiple security groups, and each security group can contain multiple security rules. You can associate multiple security groups with each Exadata Cloud Service instance, and each security group can be associated with multiple Exadata Cloud Service instances. You can dynamically enable and disable security rules by modifying the security groups that are associated with each Exadata Cloud Service instance.

To enable access to an Exadata Cloud Service instance:



Note

In the following examples, *<domain>* is the identity domain ID. An identity domain ID can be either the *IDCS GUID* that identifies the identity domain for the users within Identity Cloud Service (IDCS) or the *Identity Domain name* for a traditional Cloud Account.

To obtain the IDCS GUID

Go to the Users page in My Services dashboard and click **Identity Console**. The URL in the browser address field displays the IDCS GUID for your identity domain. For example:

```
https://idcs-
105bbbdfe5644611bf7ce04496073adf.identity.oraclecloud.com/ui/v
1/adminconsole/?root=users
```

In the above URL, `idcs-105bbbdfe5644611bf7ce04496073adf` is the IDCS GUID for your identity domain.

1. Get the service instance IDs.
   Operation: GET ServiceInstances

Example

Example request:

```
GET
/itas/<domain>/myservices/api/v1/serviceInstances?serviceDefinitionNames=Exadata&statuses=ACTIVE
```

Example payload returned for this request:

```
{
  "items": [
    {
      "id": "csi-585928949",               // Unique ServiceInstanceId
      "serviceEntitlement": {
      "id": "cesi-585927251",
      "canonicalLink": "/itas/<domain>/myservices/api/v1/serviceEntitlements/cesi-585927251"
    },
    "serviceDefinition": {
      "canonicalLink": "/itas/<domain>/myservices/api/v1/serviceDefinitions/502579309",
      "id": "502579309",
      "name": "Exadata"                // The customer is entitled to use the Exadata Service
    },
    "cloudAccount": {
      "canonicalLink": "/itas/<domain>/myservices/api/v1/cloudAccounts/cacct-fd7a122448aaaa",
      "id": "cacct-fd7a122448aaaa",
      "name": "myAccountName"
    },
    ...
    "canonicalLink": "/itas/<domain>/myservices/api/v1/serviceInstances/csi-585928949"
  }
...                                          // More Service Instances could be displayed
  ],
"canonicalLink": "/itas/<domain>/myservices/api/v1/serviceInstances",
"hasMore": false,
"limit": 25,
"offset": 0
}
```

This example payload returns the service instance ID csi-585928949, which is part of the service entitlement ID cesi-585927251.

2. Get the service configuration IDs.
   Operation: GET SIServiceConfigurations

Example

Example request, using the service instance ID csi-585928949:

```
GET /itas/<domain>/myservices/api/v1/serviceInstances/csi-585928949/serviceConfigurations
```

Example payload returned for this request:

```
{
 "canonicalLink": "/itas/<domain>/myservices/api/v1/serviceInstances/csi-
585928949/serviceConfigurations",
 "items": [
   {
     "canonicalLink": "/itas/<domain>/myservices/api/v1/serviceInstances/csi-
585928949/serviceConfigurations/Exadata",
      "exadata": {
      "bursting": {
        "canonicalLink": "/itas/<domain>/myservices/api/v1/serviceInstances/csi-
585928949/serviceConfigurations/Exadata/bursting"
      },
     "id": "Exadata",
     "securityGroupAssignments": {
     "canonicalLink": "/itas/<domain>/myservices/api/v1/serviceInstances/csi-
585928949/serviceConfigurations/Exadata/securityGroupAssignments"
     }
   },
   "id": "Exadata"
   }
 ]
}
```

This example payload shows that /itas/*<domain>*/myservices/api/v1/serviceInstances/csi-585928949/serviceConfigurations/Exadata/securityGroupAssignments is used for Exadata Firewall.

3. Get the current security groups for the service entitlement.
   Operation: GET SEExadataSecurityGroups

Example

Example request, using the service entitlement ID cesi-585927251:

```
GET /itas/<domain>/myservices/api/v1/serviceEntitlements/cesi-
585927251/serviceConfigurations/Exadata/securityGroups
```

Example payload returned for this request:

```
{
  "items": [
    {
      "id": "1",
      "customerId": "585927251",
      "name": "SecGroup 1",
      "description": "My first Security group",
      "version": 10,
      "rules": [
        {
          "direction": "ingress",
          "proto": "tcp",
          "startPort": 1159,
          "endPort": 1159,
          "ipSubnet": "0.0.0.0/0",
          "ruleInterface": "data"
        }
      ],
      "canonicalLink":
"/itas/
<domain>
/myservices/api/v1/serviceEntitlements/585927251/serviceConfigurations/Exadata/securityGroups/1"
    },
    {
      "id": "2",
      "customerId": "585927251",
      "name": " SecGroup 2",
      "description": "My second Security group",
      "version": 3,
      "rules": [
        {
          "direction": "egress",
          "proto": "tcp",
          "startPort": 8123,
          "endPort": 8123,
```

```
          "ipSubnet": "192.168.1.0/28",
          "ruleInterface": "data"
        }
      ],
      "canonicalLink":
"/itas/
<domain>
/myservices/api/v1/serviceEntitlements/585927251/serviceConfigurations/Exadata/securityGroups/2"
    }
  ],
  "canonicalLink":
"/itas/
<domain>
/myservices/api/v1/serviceEntitlements/585927251/serviceConfigurations/Exadata/securityGroups"
}
```

This example payload shows two security groups defined for the specified service entitlement ID.

4. Get the current security group assignments for the service instance
   Operation: GET SIExadataSecurityGroupAssignments

Example

Example request, using the service instance ID csi-585928949:

```
GET /itas/<domain>/myservices/api/v1/serviceInstances/csi-
585928949/serviceConfigurations/Exadata/securityGroupAssignments
```

Example payload returned for this request:

```
{
  "items": [
    {
      "id": "11",
      "securityGroup":
      {
        "id": "1",
        "canonicalLink":
"/itas/
```

```
<domain>
/myservices/api/v1/serviceEntitlements/585927251/serviceConfigurations/Exadata/securityGroups/1"
      },
      "canonicalLink": "/itas/<domain>/myservices/api/v1/serviceInstances/csi-
585928949/serviceConfigurations/Exadata/securityGroupAssignments/11"
    }
  ],
  "canonicalLink": "/itas/<domain>/myservices/api/v1/serviceInstances/csi-
585928949/serviceConfigurations/Exadata/securityGroupAssignments"
}
```

This example payload shows one security group assigned to the service instance csi-585928949.

5. Create a security group with security rules.
   Operation: POST SEExadataSecurityGroups

Example

Example request, using the service entitlement ID cesi-585927251:

```
POST /itas/<domain>/myservices/api/v1/serviceEntitlements/cesi-
585927251/serviceConfigurations/Exadata/securityGroups
{
   "customerId": "585927251",
   "name": "SecGroup 1",
   "description": "My third Security group",
   "version": 1,
   "rules": [
     {
       "direction": "ingress",
       "proto": "tcp",
       "startPort": 30,
       "endPort": 31,
       "ipSubnet": "100.100.100.255",
       "ruleInterface": "admin"
     },
     {
       "direction": "egress",
```

```
      "proto": "tcp",
      "startPort": 32,
      "endPort": 32,
      "ipSubnet": "100.100.255.0/16",
      "ruleInterface": "admin"
    }
  ]
}
```

Attributes:

Name	Description
customerId	Required: Yes String This must be the same as the *<serviceEntitlementId>*
direction	Required: Yes String Allowed values: [ingress | egress] for inbound or outbound.
proto	Required: Yes String Allowed values: [tcp | udp].
startPort	Required: Yes Integer startPort defines the beginning of a range of ports to open/white-list [0 - 65535].

Name	Description
endPort	Required: Yes Integer endPort defines the ending of a range of ports to open/white-list [0 - 65535].
ipSubnet	Required: Yes String Single IP address or range specified in CIDR notation.
ruleInterface	Required: Yes String Allowed values: [admin | client | backup] where: • admin — specifies that the rule applies to network communications over the administration network interface. The administration network is typically used to support administration tasks by using terminal sessions, monitoring agents, and so on. • client — specifies that the rule applies to network communications over the client access network interface, which is typically used by Oracle Net Services connections. • backup — specifies that the rule applies to network communications over the backup network interface, which is typically used to transport backup information to and from network-based storage that is separate from Exadata Cloud Service.

If successful, the POST request will return the unique ID of the newly created security group. For the next step, we'll assume that the newly created security group ID is 3.



Note

A security group can also be modified or deleted.
See Oracle Cloud My Services API.

6. Assign the security group to a service instance.
   Operation: POST SIExadataSecurityGroupAssignments

Example

Example request, using the service instance csi-585928949 and the security group ID 3:

```
POST /itas/<domain>/myservices/api/v1/serviceInstances/csi-
585928949/serviceConfigurations/Exadata/securityGroupAssignments

{
   "securityGroup": {
      "id": "3",
      "customerId": "585927251",
      "canonicalLink":
"/itas/
<domain>
/myservices/api/v1/serviceEntitlements/585927251/serviceConfigurations/Exadata/securityGroups/3"
   }
}
```

Attributes:

Name	Description
customerId	Required: Yes String This must be the same as the serviceEntitlementId.

If successful, the POST request will return the unique Id of the newly created security group assignment.



Note

A security group assignment can also be deleted. See Oracle Cloud My Services API.

You can now verify all your security groups and assignments. See:

- Get the current security groups for the service entitlement.
- Get the current security group assignments for the service instance .

Exadata Scaling with Bursting

You can temporarily modify the capacity of your Exadata environment by configuring bursting. Bursting is a method you can use to scale Exadata Cloud Service non-metered instances within an Exadata system.

To scale up your non-metered instances, increase the number of compute nodes by modifying the `burstOcpu` attribute of the host. When you no longer need the additional nodes, update the `burstOcpu` attribute back to its original setting.



Note

In the following examples, *<domain>* is the identity domain ID. An identity domain ID can be either the *IDCS GUID* that identifies the identity domain for the users within Identity Cloud Service (IDCS) or the *Identity Domain name* for a traditional Cloud Account.

To obtain the IDCS GUID

Go to the Users page in My Services dashboard and click **Identity Console**. The URL in the browser address field displays the IDCS GUID for your identity domain. For example:

```
https://idcs-
105bbbdfe5644611bf7ce04496073adf.identity.oraclecloud.com/ui/v
1/adminconsole/?root=users
```

In the above URL, `idcs-105bbbdfe5644611bf7ce04496073adf` is the IDCS GUID for your identity domain.

1. Get the service instance IDs.
   Operation: GET ServiceInstances

Example

Example request:

```
GET
/itas/<domain>/myservices/api/v1/serviceInstances?serviceDefinitionNames=Exadata&statuses=ACTIVE
```

Example payload returned for this request:

```
{
  "items": [
    {
      "id": "csi-585928949",               // Unique ServiceInstanceId
      "serviceEntitlement": {
      "id": "cesi-585927251",
      "canonicalLink": "/itas/<domain>/myservices/api/v1/serviceEntitlements/cesi-585927251"
    },
    "serviceDefinition": {
      "canonicalLink": "/itas/<domain>/myservices/api/v1/serviceDefinitions/502579309",
      "id": "502579309",
      "name": "Exadata"                // The customer is entitled to use the Exadata Service
    },
    "cloudAccount": {
      "canonicalLink": "/itas/<domain>/myservices/api/v1/cloudAccounts/cacct-fd7a122448aaaa",
      "id": "cacct-fd7a122448aaaa",
      "name": "myAccountName"
    },
    ...
    "canonicalLink": "/itas/<domain>/myservices/api/v1/serviceInstances/csi-585928949"
  }
...                                          // More Service Instances could be displayed
  ],
"canonicalLink": "/itas/<domain>/myservices/api/v1/serviceInstances",
"hasMore": false,
"limit": 25,
"offset": 0
}
```

This example payload returns the service instance ID csi-585928949.

2. Get the service configuration IDs.
   Operation: GET SIServiceConfigurations

Example

Example request, using the service instance ID csi-585928949:

```
GET /itas/<domain>/myservices/api/v1/serviceInstances/csi-585928949/serviceConfigurations
```

Example payload returned for this request:

```
{
 "canonicalLink": "/itas/<domain>/myservices/api/v1/serviceInstances/csi-
585928949/serviceConfigurations",
 "items": [
   {
     "canonicalLink": "/itas/<domain>/myservices/api/v1/serviceInstances/csi-
585928949/serviceConfigurations/Exadata",
      "exadata": {
      "bursting": {
        "canonicalLink": "/itas/<domain>/myservices/api/v1/serviceInstances/csi-
585928949/serviceConfigurations/Exadata/bursting"
      },
     "id": "Exadata",
     "securityGroupAssignments": {
     "canonicalLink": "/itas/<domain>/myservices/api/v1/serviceInstances/csi-
585928949/serviceConfigurations/Exadata/securityGroupAssignments"
     }
   },
   "id": "Exadata"
   }
 ]
}
```

This example payload shows that /itas/*<domain>*/myservices/api/v1/serviceInstances/csi-585928949/serviceConfigurations/Exadata/securityGroupAssignments is used for Bursting.

3. Get the current compute node configuration.
   Operation: GET SIExadataBursting

Example

Example request, using the service instance ID csi-585928949:

```
GET /itas/<domain>/myservices/api/v1/serviceInstances/csi-
585928949/serviceConfigurations/Exadata/bursting
```

Example payload returned for this request:

```
{
   "ocpuOpInProgress": false,
   "exaunitId": 50,
   "ocpuAllocations": [
       {
          "hostName": "host1.oraclecloud.com",
          "subscriptionOcpu": 11,
          "meteredOcpu": 0,
          "burstOcpu": 0,                                  // Current Burst value
          "minOcpu": 11,
          "maxOcpu": 42,
          "maxBurstOcpu": 11,
          "maxSubOcpu": 38,
          "maxMetOcpu": 0
       },
       {
          "hostName": "host2.oraclecloud.com",
          "subscriptionOcpu": 11,
          "meteredOcpu": 0,
          "burstOcpu": 0,                                  // Current Burst value
          "minOcpu": 11,
          "maxOcpu": 42,
          "maxBurstOcpu": 11,
          "maxSubOcpu": 38,
          "maxMetOcpu": 0
       }
       ],
   "status": 200,
   "op": "exaunit_coreinfo",
   "additionalNumOfCores": "0",
   "additionalNumOfCoresHourly": "0",
   "coreBursting": "Y"
}
```

4. Modify the values for `burstOcpu`.
   Operation: PUT SIExadataBursting
   You can modify `burstOcpu` to a value that is up to the value of `maxBurstOcpu`. This example adds two compute nodes to each host.

Example

Example request, using the service instance csi-585928949:

```
PUT /itas/<domain>/myservices/api/v1/serviceInstances/csi-
585928949/serviceConfigurations/Exadata/bursting/
{
   "ocpuOpInProgress": false,
   "exaunitId": 50,
   "ocpuAllocations": [
       {
          "hostName": "host1.oraclecloud.com",
          "subscriptionOcpu": 11,
          "meteredOcpu": 0,
          "burstOcpu": 2,
          "minOcpu": 11,
          "maxOcpu": 42,
          "maxBurstOcpu": 11,
          "maxSubOcpu": 38,
          "maxMetOcpu": 0
       },
       {
          "hostName": "host2.oraclecloud.com",
          "subscriptionOcpu": 11,
          "meteredOcpu": 0,
          "burstOcpu": 2,
          "minOcpu": 11,
          "maxOcpu": 42,
          "maxBurstOcpu": 11,
          "maxSubOcpu": 38,
          "maxMetOcpu": 0
       }
       ]
}
```

Attributes:

Name	Description
burstOcpu	Required: Yes Type: Integer, Minimum Value: 0, Maximum Value: maxBurstOcpu Number of additional cores



Note

This action may take a few minutes to complete.

5. Verify the new compute node configuration.
   Operation: GET SIExadataBursting

Example

Example request, using the service instance ID csi-585928949:

```
GET /itas/<domain>/myservices/api/v1/serviceInstances/csi-
585928949/serviceConfigurations/Exadata/bursting
```

Example payload returned for this request:

```
{
   "ocpuOpInProgress": false,
   "exaunitId": 50,
   "ocpuAllocations": [
       {
          "hostName": "host1.oraclecloud.com",
          "subscriptionOcpu": 11,
          "meteredOcpu": 0,
          "burstOcpu": 2,                    // New Burst value
          "minOcpu": 11,
          "maxOcpu": 42,
          "maxBurstOcpu": 11,
          "maxSubOcpu": 38,
          "maxMetOcpu": 0
```

```
      },
      {
         "hostName": "host2.oraclecloud.com",
         "subscriptionOcpu": 11,
         "meteredOcpu": 0,
         "burstOcpu": 2,                    // New Burst value
         "minOcpu": 11,
         "maxOcpu": 42,
         "maxBurstOcpu": 11,
         "maxSubOcpu": 38,
         "maxMetOcpu": 0
      }
      ],
  "status": 200,
  "op": "exaunit_coreinfo",
  "additionalNumOfCores": "0",
  "additionalNumOfCoresHourly": "0",
  "coreBursting": "Y"
}
```

Managing Exadata Instances

The following procedures walk you through creating, modifying, and deleting Exadata instances used with the Oracle Cloud My Services API.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.



Important

These procedures are for use with Oracle Database Exadata Cloud Service. For more information, see Administering Oracle Database Exadata Cloud Service. These procedures **DO NOT** apply to Exadata DB Systems available in Oracle Cloud Infrastructure.

Prerequisites

Before you can manage Exadata instances, you need to:

- Subscribe to an Oracle Cloud service
- Obtain account credentials with required roles assigned
- Determine your API endpoint

To subscribe to an Oracle Cloud service

To access Oracle Cloud My Services API, you must request a trial or paid subscription to an Oracle Cloud service.

To obtain account credentials and role assignments

Ask your account administrator for the following items to access Oracle Cloud My Services API:

- Account credentials:
  - User name and password
  - Identity domain ID

An identity domain ID can be either the *IDCS GUID* that identifies the identity domain for the users within Identity Cloud Service (IDCS) or the *Identity Domain name* for a traditional Cloud Account.

- Required roles assigned to above user name

To determine your API endpoint

Insert the identity domain ID provided by the account administrator (*<domain>*) between `/itas/` and `/myservices/`.

Example:

```
https://itra.oraclecloud.com/itas/<domain>/myservices/api/v1/serviceEntitlements
```

Creating Exadata Instances

This section covers how to create a basic Exadata instance, an instance with custom IP network configuration, and an instance with multi-VM support.

To create a basic Exadata instance

Post a request with the required payload to create a new instance for a given service entitlement (Exadata in our case).

In the following example, *<domain>* is the identity domain ID.

```
POST /itas/<domain>/myservices/api/v1/operations
{
  "operationItems": [
    {
      "attributes": [
        {
          "name": "requestPayload.name",
          "value": "newinstanceName"
        },
        {
          "name": "requestPayload.serviceEntitlementId",
```

```
        "value": "500073421"
      },
      {
        "name": "requestPayload.size",
        "value": "CUSTOM"
      },
      {
        "name": "requestPayload.serviceType",
        "value": "Exadata"
      },
      {
        "name": "requestPayload.adminUserName",
        "value": "john.smith@example.com"
      },
      {
        "name": "requestPayload.adminEmail",
        "value": "john.smith@example.com"
      },
      {
        "name": "requestPayload.adminFirstName",
        "value": "John"
      },
      {
        "name": "requestPayload.adminLastName",
        "value": "Smith"
      },
      {
        "name": "requestPayload.invokerAdminUserName",
        "value": "john.smith@example.com"
      },
      {
        "name": "requestPayload.invokerAdminEmail",
        "value": "john.smith@example.com"
      },
      {
        "name": "requestPayload.invokerAdminFirstName",
        "value": "John"
      },
      {
        "name": "requestPayload.invokerAdminLastName",
        "value": "Smith"
```

```
        },
        {
          "name": "requestPayload.customAttributes.ExaUnitName",
          "value": "systemname"
        },
        {
          "name": "requestPayload.customAttributes.CreateSparse",
          "value": "N"
        },
        {
          "name": "requestPayload.customAttributes.BackupToDisk",
          "value": "N"
        },
        {
          "name": "requestPayload.customAttributes.isBYOL",
          "value": "N"
        },
        {
          "name": "requestPayload.customAttributes.PickRackSize",
          "value": "Quarter Rack"
        },
        {
          "name": "requestPayload.customAttributes.SELECTED_DC_ID",
          "value": "US001"
        }
      ],
        "operationItemDefinition": {
        "id": "CIM-Exadata-CUSTOM-PRODUCTION-CREATE"
      }
    }
  ]
}
```

ATTRIBUTES

Name	Description
requestPayload.name	Required: Yes Type: String Name of the Exadata instance. This name: • Must not exceed 25 characters. • Must start with a letter. • Must contain only lower case letters and numbers. • Must not contain spaces or any other special characters. • Must be unique within the identity domain.
requestPayload. serviceEntitlementId	Required: Yes Type: String Service Entitlement for the Exadata instance. See "Exadata Service Entitlement discovery". Note that any "cesi-" or "sub-" prefix should not be included.
requestPayload. customAttributes. ExaUnitName	Required: Yes Type: String A name for your Exadata Database Machine environment. This name is also used as the cluster name for the Oracle Grid Infrastructure installation.

Name	Description
requestPayload. customAttributes. CreateSparse	Required: Yes Type: String "Y" to create a disk group that is based on sparse grid disks, else "N". You must select this option to enable Exadata Cloud Service snapshots. Exadata snapshots enable space-efficient clones of Oracle databases that can be created and destroyed very quickly and easily.
requestPayload. customAttributes. BackupToDisk	Required: Yes Type: String "Y" to use "Database backups on Exadata Storage", else "N". This option configures the Exadata storage to enable local database backups on Exadata storage.
requestPayload. customAttributes. isBYOL	Required: Yes Type: String "Y" to indicate that the Exadata Cloud Service instance uses Oracle Database licenses that are provided by you rather than licenses that are provided are part of the service subscription, else "N". This option only affects the billing that is associated with the service instance. It has no effect on the technical configuration of the Exadata Cloud Service instance.

Name	Description
requestPayload. customAttributes. PickRackSize	Required: Yes Type: String Specify the rack configuration for your service instance. Exact allowed values depend on your purchase. Typical values are like "Full Rack", "Half Rack", "Quarter Rack" or "Eighth Rack".
requestPayload. customAttributes. SELECTED_DC_ID	Required: Yes Type: String Data center that will host your Exadata Cloud Service instance. See "Exadata Service Entitlement discovery" to obtain the Eligible Data Center IDs.

To create an Exadata instance with custom IP network configuration

Post a request with the attributes ClientNetwork and BackupNetwork as part of the payload. The following example includes these optional attributes as well as required attributes.

In the following example, *<domain>* is the identity domain ID.

```
POST /itas/<domain>/myservices/api/v1/operations
{
  "operationItems": [
    {
      "attributes": [
        {
          "name": "requestPayload.name",
          "value": "newinstanceName"
        },
        {
          "name": "requestPayload.serviceEntitlementId",
          "value": "500073421"
        },
```

```
        {
          "name": "requestPayload.size",
          "value": "CUSTOM"
        },
        {
          "name": "requestPayload.serviceType",
          "value": "Exadata"
        },
        {
          "name": "requestPayload.adminUserName",
          "value": "john.smith@example.com"
        },
        {
          "name": "requestPayload.adminEmail",
          "value": "john.smith@example.com"
        },
        {
          "name": "requestPayload.adminFirstName",
          "value": "John"
        },
        {
          "name": "requestPayload.adminLastName",
          "value": "Smith"
        },
        {
          "name": "requestPayload.invokerAdminUserName",
          "value": "john.smith@example.com"
        },
        {
          "name": "requestPayload.invokerAdminEmail",
          "value": "john.smith@example.com"
        },
        {
          "name": "requestPayload.invokerAdminFirstName",
          "value": "John"
        },
        {
          "name": "requestPayload.invokerAdminLastName",
          "value": "Smith"
        },
        {
```

```
          "name": "requestPayload.customAttributes.ExaUnitName",
          "value": "systemname"
        },
        {
          "name": "requestPayload.customAttributes.CreateSparse",
          "value": "N"
        },
        {
          "name": "requestPayload.customAttributes.BackupToDisk",
          "value": "N"
        },
        {
          "name": "requestPayload.customAttributes.isBYOL",
          "value": "N"
        },
        {
          "name": "requestPayload.customAttributes.PickRackSize",
          "value": "Quarter Rack"
        },
        {
          "name": "requestPayload.customAttributes.SELECTED_DC_ID",
          "value": "US001"
        }
        {
          "name": "requestPayload.customAttributes.ClientNetwork",
          "value": "/root/root/1/ipnetwork1"
        },
        {
          "name": "requestPayload.customAttributes.BackupNetwork",
          "value": "/root/root/1/ipnetwork2"
        }
      ],
        "operationItemDefinition": {
        "id": "CIM-Exadata-CUSTOM-PRODUCTION-CREATE"
      }
    }
  ]
}
```

ATTRIBUTES

Name	Description
requestPayload. customAttributes. ClientNetwork	Required: Yes Type: Url IP network definitions for the network that is primarily used for client access to the database servers. Applications typically access databases on Exadata Cloud Service through this network using Oracle Net Services in conjunction with Single Client Access Name (SCAN) and Oracle RAC Virtual IP (VIP) interfaces.
requestPayload. customAttributes. BackupNetwork	Required: Yes Type: Url IP network definitions for the network that is typically used to access the database servers for various purposes, including backups and bulk data transfers.

To create an Exadata instance with multi-VM support

If your Exadata system environment is enabled to support multiple virtual machine (VM) clusters, then you can define up to eight clusters and specify how the overall Exadata system resources are allocated to them.

In a configuration with multiple VM clusters, each VM cluster is allocated a dedicated portion of the overall Exadata system resources, with no over-provisioning or resource sharing. On the compute nodes, a separate VM is defined for each VM cluster, and each VM is allocated a dedicated portion of the available compute node CPU, memory, and local disk resources. Each VM cluster is also allocated a dedicated portion of the overall Exadata storage.

Post a request with the attributes EXAUNIT_ALLOCATIONS and MULTIVM_ENABLED as part of the payload. The following example includes these optional attributes as well as required attributes.

In the following example, *<domain>* is the identity domain ID and *<base64_encoded_string>* is a base64 encoding of the payload following the example.

Example payload for request:

```
POST /itas/<domain>/myservices/api/v1/operations
{
  "operationItems": [
    {
      "attributes": [
        {
          "name": "requestPayload.name",
          "value": "newinstanceName"
        },
        {
          "name": "requestPayload.serviceEntitlementId",
          "value": "500073421"
        },
        {
          "name": "requestPayload.size",
          "value": "CUSTOM"
        },
        {
          "name": "requestPayload.serviceType",
          "value": "Exadata"
        },
        {
          "name": "requestPayload.adminUserName",
          "value": "john.smith@example.com"
        },
        {
          "name": "requestPayload.adminEmail",
          "value": "john.smith@example.com"
        },
        {
          "name": "requestPayload.adminFirstName",
          "value": "John"
```

```
    },
    {
      "name": "requestPayload.adminLastName",
      "value": "Smith"
    },
    {
      "name": "requestPayload.invokerAdminUserName",
      "value": "john.smith@example.com"
    },
    {
      "name": "requestPayload.invokerAdminEmail",
      "value": "john.smith@example.com"
    },
    {
      "name": "requestPayload.invokerAdminFirstName",
      "value": "John"
    },
    {
      "name": "requestPayload.invokerAdminLastName",
      "value": "Smith"
    },
    {
      "name": "requestPayload.customAttributes.ExaUnitName",
      "value": "systemname"
    },
    {
      "name": "requestPayload.customAttributes.CreateSparse",
      "value": "N"
    },
    {
      "name": "requestPayload.customAttributes.BackupToDisk",
      "value": "N"
    },
    {
      "name": "requestPayload.customAttributes.isBYOL",
      "value": "N"
    },
    {
      "name": "requestPayload.customAttributes.PickRackSize",
      "value": "Quarter Rack"
    },
```

```
        {
          "name": "requestPayload.customAttributes.SELECTED_DC_ID",
          "value": "US001"
        }
        {
          "name": "requestPayload.customAttributes.EXAUNIT_ALLOCATIONS",
          "value": "<base64_encoded_string>"
        },
        {
          "name": "requestPayload.customAttributes.MULTIVM_ENABLED",
          "value": "true"
        }
      ],
        "operationItemDefinition": {
        "id": "CIM-Exadata-CUSTOM-PRODUCTION-CREATE"
      }
    }
  ]
}
```

Payload for *<base64_encoded_string>*:

```
{
  ExaunitProperties: [
    {name:requestId, value:27ac0ee3-0c72-4493-b02b-40038f07d2a0},
    {name:Operation, value:AddCluster},
    {name:TotalNumOfCoresForCluster, value:4},
    {name:TotalMemoryInGb, value:30},
    {name:StorageInTb, value:3},
    {name:OracleHomeDiskSizeInGb, value:60},
    {name:ClientNetwork, value:/root/root/1/ipnetwork1},  // Only if Higgs is also required
    {name:BackupNetwork, value:/root/root/1/ipnetwork2},  // Only if Higgs is also required
    {name:ExaUnitName, value:systemname},
    {name:CreateSparse, value:N},
    {name:BackupToDisk, value:N}
  ]
}
```

ATTRIBUTES

Name	Description
requestId	Required: Optional Type: String Unique UUID
TotalNumOfCores ForCluster	Required: Yes Type: String The number of CPU cores that are allocated to the VM cluster. This is the total number of CPU cores that are allocated evenly across all of the compute nodes in the VM cluster. Must be a multiple of numComputes as returned by a call to ecra/endpoint/clustershapes.
TotalMemoryInGb	Required: Yes Type: String The amount of memory (in GB) that is allocated to the VM cluster. This is the total amount of memory that is allocated evenly across all of the compute nodes in the VM cluster. Must be a multiple of numComputes as returned by a call to ecra/endpoint/clustershapes.

Name	Description
StorageInTb	Required: Yes Type: String The total amount of Exadata storage (in TB) that is allocated to the VM cluster. This storage is allocated evenly from all of the Exadata Storage Servers.
OracleHomeDiskSize InGb	Required: Yes Type: String The amount of local disk storage (in GB) that is allocated to each database server in the first VM cluster.

Modifying Exadata Instances

This section covers how to add a cluster to an existing instance, reshape a cluster, and delete a cluster.

To add a cluster to an existing instance

Post a request with the operationItemDefinition of CIM-Exadata-CUSTOM-PRODUCTION-UPDATE and a base64 encoding of a payload that includes the Operation value of AddCluster.

In the following example, *<domain>* is the identity domain ID, *<instanceId>* and *<serviceEntitlementId>* are returned from iTAS serviceInstances, and *<base64_encoded_string>* is a base64 encoding of the payload following the example.

Example payload for request:

```
POST /itas/<domain>/myservices/api/v1/operations HTTP/1.1
{
  "operationItems": [
    {
      "attributes": [
```

```
        {
          "name": "instanceId",
          "value": "<instanceId>"
        },
        {
          "name": "requestPayload.serviceEntitlementId",
          "value": "<serviceEntitlementId>"
        },
        {
          "name": "requestPayload.size",
          "value": "CUSTOM"
        },
        {
          "name": "requestPayload.serviceType",
          "value": "Exadata"
        },
        {
          "name": "requestPayload.customAttributes.EXAUNIT_ALLOCATIONS",
          "value": "<base64_encoded_string>"
        },
        {
          "name": "requestPayload.customAttributes. MULTIVM_ENABLED",
          "value": "true"
        }
      ],
        "operationItemDefinition": {
        "id": "CIM-Exadata-CUSTOM-PRODUCTION-UPDATE"
      }
    }
  ]
}
```

Payload for *<base64_encoded_string>*:

```
{
  ExaunitProperties: [
    {name:requestId, value:27ac0ee3-0c72-4493-b02b-40038f07d2a0},
    {name:Operation, value:AddCluster},
    {name:TotalNumOfCoresForCluster, value:4},
    {name:TotalMemoryInGb, value:30},
    {name:StorageInTb, value:3},
    {name:OracleHomeDiskSizeInGb, value:60},
```

```
    {name:ClientNetwork, value:/root/root/1/ipnetwork1},  // Only if Higgs is also required
    {name:BackupNetwork, value:/root/root/1/ipnetwork2},  // Only if Higgs is also required
    {name:ExaUnitName, value:Cluster2},
    {name:CreateSparse, value:N},
    {name:BackupToDisk, value:N}
  ]
}
```

To reshape a cluster

Post a request with the operationItemDefinition of CIM-Exadata-CUSTOM-PRODUCTION-UPDATE and a base64 encoding of a payload that includes the Operation value of ReshapeCluster.

In the following example, *<domain>* is the identity domain ID and *<base64_encoded_string>* is a base64 encoding of the payload following the example.

Example payload for request:

```
POST /itas/<domain>/myservices/api/v1/operations HTTP/1.1
{
  "operationItems": [
    {
      "attributes": [
        {
          "name": "instanceId",
          "value": "500076173"
        },
        {
          "name": "requestPayload.serviceEntitlementId",
          "value": "500073421"
        },
        {
          "name": "requestPayload.size",
          "value": "CUSTOM"
        },
        {
          "name": "requestPayload.serviceType",
          "value": "Exadata"
        },
```

```
        {
          "name": "requestPayload.customAttributes.EXAUNIT_ALLOCATIONS",
          "value": "<base64_encoded_string>"
        },
        {
          "name": "requestPayload.customAttributes. MULTIVM_ENABLED",
          "value": "true"
        }
      ],
        "operationItemDefinition": {
        "id": "CIM-Exadata-CUSTOM-PRODUCTION-UPDATE"
      }
    }
  ]
}
```

Payload for *<base64_encoded_string>*:

```
{
  ExaunitProperties: [
    {name:requestId, value:27ac0ee3-0c72-4493-b02b-40038f07d2a0},
       {name:ExaunitID, value:1},              // From ecra/endpoint/exaservice/{serviceInstance}/resourceinfo
    {name:Operation, value:ReshapeCluster},
    {name:TotalNumOfCoresForCluster, value:10},
    {name:TotalMemoryInGb, value:10},
    {name:StorageInTb, value:4},
    {name:OhomePartitionInGB, value:100},
    {name:ClientNetwork, value:/root/root/1/ipnetwork1},  // Only if Higgs is also required
    {name:BackupNetwork, value:/root/root/1/ipnetwork2}   // Only if Higgs is also required
  ]
}
```



Important

- Only one attribute can be modified per Reshape request. The payload should contain only the modified attribute. Example:

```
{ExaunitProperties:
[{name:Operation,value:ReshapeCluster},
{name:ExaunitID,value:5},
{name:TotalNumOfCoresForCluster,value:6}]}
```

- When doing a Reshape with the `OracleHomeDiskSizeInGb` attribute, use the name `OhomePartitionInGB`.
- The value for `TotalNumOfCoresForCluster` must be a multiple of `numComputes` as returned by a call to `ecra/endpoint/clustershapes`.
- The value for `TotalMemoryInGb` must be a multiple of `numComputes` as returned by a call to `ecra/endpoint/clustershapes`.

To delete a cluster

Post a request with the operationItemDefinition of CIM-Exadata-CUSTOM-PRODUCTION-UPDATE and a base64 encoding of a payload that includes the Operation value of DeleteCluster.

In the following example, *<domain>* is the identity domain ID and *<base64_encoded_string>* is a base64 encoding of the payload following the example.

Example payload for request:

```
POST /itas/<domain>/myservices/api/v1/operations HTTP/1.1
{
```

```
  "operationItems": [
    {
      "attributes": [
        {
          "name": "instanceId",
          "value": "500076173"
        },
        {
          "name": "requestPayload.serviceEntitlementId",
          "value": "500073421"
        },
        {
          "name": "requestPayload.size",
          "value": "CUSTOM"
        },
        {
          "name": "requestPayload.serviceType",
          "value": "Exadata"
        },
        {
          "name": "requestPayload.customAttributes.EXAUNIT_ALLOCATIONS",
          "value": "<base64_encoded_string>"
        },
        {
          "name": "requestPayload.customAttributes. MULTIVM_ENABLED",
          "value": "true"
        }
      ],
        "operationItemDefinition": {
        "id": "CIM-Exadata-CUSTOM-PRODUCTION-UPDATE"
      }
    }
  ]
}
```

Payload for *<base64_encoded_string>*:

```
{
  ExaunitProperties: [
    {name:requestId, value:27ac0ee3-0c72-4493-b02b-40038f07d202},  // Optional
       {name:ExaunitID, value:2},
    {name:Operation, value:DeleteCluster}
```

```
  ]
}
```

Deleting Exadata Instances

This section covers how to delete Exadata instances.



Important

Delete all existing multi-VM clusters before deleting the Exadata instance. Following this guidance prevents the instance ending up in an invalid state.

To delete an instance

Post a request with the operationItemDefinition of CIM-Exadata-CUSTOM-PRODUCTION-DELETE.

In the following example, *<domain>* is the identity domain ID.

Example payload for request:

```
POST /itas/<domain>/myservices/api/v1/operations HTTP/1.1
{
  "operationItems": [
    {
      "attributes": [
        {
          "name": "instanceId",
          "value": "500076173"
        },
        {
          "name": "requestPayload.serviceEntitlementId",
          "value": "500073421"
        },
        {
          "name": "requestPayload.serviceType",
```

```
          "value": "Exadata"
        }
      ],
        "operationItemDefinition": {
        "id": "CIM-Exadata-CUSTOM-PRODUCTION-DELETE"
      }
    }
  ]
}
```

Discovering Entitlements and Instances

This section describes how to discover service entitlements and service instances.

To discover service entitlements

Send the following request:

```
GET /itas/<domain>/myservices/api/v1/serviceEntitlements?serviceDefinitionNames=Exadata
```

Example payload returned for this request:

```
{
    "items": [
      {
        "id": "cesi-585927251",               // Unique ServiceEntitlementId
        "serviceDefinition": {
          "canonicalLink": "/itas/a517289/myservices/api/v1/serviceDefinitions/502579309",
          "id": "502579309",
          "name": "Exadata"                 // The customer is entitled to use the Exadata Service
        },
        "status": "ACTIVE",
        ...
        "canonicalLink": "/itas/a517289/myservices/api/v1/serviceInstances/csi-585928949"
      }
    ...                                              // More Service Entitlements could be displayed
    ],
  "canonicalLink": "/itas/<domain>/myservices/api/v1/serviceEntitlements",
  "hasMore": false,
  "limit": 25,
```

```
  "offset": 0
}
```

Eligible Data Centers:

Use:

```
/itas/<domain>/myservices/api/v1/serviceEntitlements/
{ServiceEntitlementId}?expands=serviceInstancesEligibleDataCenters
```

where `{ServiceEntitlementId}` is a service entitlement ID such as `cesi-500074601`. This will provide additional information such as:

```
  "serviceInstancesEligibleDataCenters": [
    {
      "id": "US001"
    }
  ],
```

To discover service instances

Send the following request:

```
GET /<domain>/myservices/api/v1/serviceInstances?serviceDefinitionNames=Exadata
```

Example payload returned for this request:

```
{
    "items": [
      {
        "id": "csi-585928949",               // Unique ServiceInstanceId
        "serviceEntitlement": {
          "id": "cesi-585927251",          // Related ServiceEntitlementId
          "canonicalLink": "/itas/a517289/myservices/api/v1/serviceEntitlements/cesi-585927251"
        },
        "serviceDefinition": {
          "canonicalLink": "/itas/a517289/myservices/api/v1/serviceDefinitions/502579309",
          "id": "502579309",
          "name": "Exadata"                 // The customer is entitled to use the Exadata Service
        },
        ...
        "canonicalLink": "/itas/a517289/myservices/api/v1/serviceInstances/csi-585928949"
```

```
      }
    ...                                                  // More Service Entitlements could be displayed
    ],
  "canonicalLink": "/itas/<domain>/myservices/api/v1/serviceEntitlements",
  "hasMore": false,
  "limit": 25,
  "offset": 0
}
```

Using Access Token Authorization with My Services API

This topic explains how to set up and use access token authorization with the Oracle Cloud My Services API. Access token authorization allows a developer to access programmatic endpoints (APIs) to obtain some information (for example, entitlements, instances, or metering data) for your cloud account.

About Access Tokens

An access token contains the information required to allow a developer to access information on your cloud account. A developer presents the token when making API calls. The allowed actions and endpoints depend on the scopes (permissions) that you select when you generate the token. An access token is valid for about an hour.

A refresh token allows the developer to generate a new access token without having to contact an administrator. A refresh token is valid for about one year.

Process Overview

Setup steps for the Administrator:

1. Create an Identity Cloud Service client application with the specific privileges you want to grant to developers.
2. Generate an access token that contains the required privileges for the intended developer.

3. Provide the access token and required information to the developer.
4. Configure Identity Cloud Service for access token validation.

Steps for developer to use the token:

1. Issue requests against My Services API endpoints. Include the access token for the authorization parameter.
2. When the access token expires, refresh the access token without administrator intervention until the privilege is terminated.

Administrator Tasks to Set Up Token Validation

Perform the following tasks to enable developer access with an access token:

Create the IDCS client application

1. Sign in to Identity Cloud Services as an Administrator and go to the administration console. See How to Access Oracle Identity Cloud Service if you need help signing in.
2. Click the **Applications** tile. A list of the applications is displayed.
3. Click **+ Add** to create a new application.
4. Click **Trusted Application** as the type of application.
5. In the **App Details** section, enter a **Name** and **Description** and then click **Next**.
6. In the **Client** section:
    a. Select **Configure this application as a client now**.
    b. Under **Authorization**, for **Allowed grant types**, select the following options:
        - **JWT Assertion**
        - **Refresh Token**
7. Under **Accessing APIs from Other Applications**, from the **Trust Scope** list, select **Allowed scopes**.
8. Under **Allowed Scopes** click **+ Add**.

9. In the **Add Scope** dialog, click the arrow next to **CloudPortalResourceApp** in the list of App.
10. Select the box next to each authorization that you might want to give the developers to whom you will provide an Access Token. (The permissions are assigned in another step.)
11. Click **Add** to close the dialog. Your selections are displayed.
12. Click **Next**.
13. In the **Resources** section, accept the default and click **Next**.
14. In the **Web Tier Policy** section, accept the default and click **Next**.
15. In the **Authorization** section, click **Finish**.
    The **Application Added** notification displays the new Client ID and Client Secret for the application.

> **Important**
>
> Copy and store the Client ID and Client Secret in a safe place and then click **Close**. The Client ID and Client Secret are credentials that are specific to the application that you just created. You will need these credentials later.

16. To complete the creation process, click **Activate** at the top of the page.

Generate an access token

1. Navigate to the IDCS application that you created in the preceding task and select the **Details** tab.
2. Click **Generate Access Token**.
3. On the **Generate Token** dialog, select **Customized Scopes**, then select **Invokes**

Other APIs.

4. Select the scopes that you want to give to the developer who will receive this access token.



Note

Oracle recommends that you provide only the minimum required privileges.

5. Select **Include Refresh Token**.
6. Click **Download Token**. Your browser will prompt you to download a token file (.tok). The token file contains an access token and a refresh token.
7. Provide this file to the developer.

Send the access information to a developer

To call API endpoints, the developer needs:

- A token file that you generated.
- The Client ID and Client Secret for the IDCS application used to generate the token file. The Client ID and Secret are required for the developer to generate a new access token from the refresh token.
- The endpoints for the APIs.
    - End points related to the itas:myservices scopes are: `https://itra.oraclecloud.com/itas/<tenant-IDCS-ID>/myservices/api/v1`
    - End points related to the itas:metering scopes are: `https://itra.oraclecloud.com/metering/api/v1`

Make sure that you send the above information in a secure way. If you think that this information has been compromised, see Revoking a Developer's Ability to Refresh Access Tokens.

Configure Identity Cloud Service for access token validation

To allow clients to access the tenant signing certificate without logging in to Oracle Identity Cloud Service:

1. Sign in to the Oracle Identity Cloud Services admin console. See How to Access Oracle Identity Cloud Service if you need help signing in.
2. Open the navigation menu. Under **Settings** select **Default Settings**.
3. Set the **Access Signing Certificate** toggle button to on.

Using the Access Token

The token file has a .tok extension. The file contains the access token and the refresh token. The content looks like:

```
{"app_access_token":"eyJ4N...aabb...CpNwA","refresh_token":"AQID...9NCA="}
```

To use the token with the My Services API:

1. Open the token file.
2. Issue a request to a valid endpoint, inserting the access token for the `Authorization` parameter.
   For example:

   ```
   curl -X GET  https://itra.oraclecloud.com/itas/<tenant-IDCS-ID>/myservices/api/v1/serviceEntitlements   -H 'Authorization: Bearer eyJ4N...aabb...CpNwA'
   ```

Requesting a New Access Token from a Refresh Token

An access token is valid for about one hour. When the token is no longer valid you will get a 401 response code and an Error Message ("errorMessage") value containing "Expired".

You can generate a new short-lived access token from the refresh token. You'll need the Client ID and Client Secret to generate the new token. You can only generate tokens with the same or lower access (scopes) as your original token.

Example using the curl command:

```
curl -i -H 'Authorization: Basic <base64Encoded clientid:secret>' -H 'Content-Type: application/x-www-form-urlencoded;charset=UTF-8' --request POST https://<tenant-IDCS-ID>/oauth2/v1/token -d 'grant_type=refresh_token&refresh_token=<refresh-token>'
```

Using the sample token file from the previous section, the value for *<refresh-token>* would be `AQID...9NCA=`.

Sample response:

```
{ "access_token": "eyJraWQiO....2nqA", "token_type": "Bearer", "expires_in": 3600, "refresh_token": "AQIDBAUn…VkxNCB7djF9NCA=" }
```



Note

When a developer generates a new access token and refresh token, the previous refresh token becomes invalid.

Revoking a Developer's Ability to Refresh Access Tokens

If you need to revoke a developer's ability to refresh access tokens, you can either invalidate the existing refresh token by generating a new Client Secret for the token; or, you can temporarily revoke access by deactivating the application.



Important

Taking either of these actions will terminate or suspend the ability of all developers using the current Client Secret or application. When generating tokens for multiple developers, consider creating more than one IDCS application to isolate developers from each other.

To terminate a developer's ability to refresh their access token

1. Sign in to Identity Cloud Services as an Administrator and go to the administration console. See How to Access Oracle Identity Cloud Service if you need help signing in.
2. Click the **Applications** tile. A list of the applications is displayed.
3. Click the application used to generate the token to view its details.
4. Click **Configuration**.
5. Under **General Information**, next to **Client Secret**, click **Regenerate** to generate a new Client Secret.

To restore the ability for the developer to generate an access token from a refresh token, generate a new access token. Then provide the token along with the new Client Secret to the developer.

To temporarily suspend a developer's ability to refresh their access token

1. Sign in to Identity Cloud Services as an Administrator and go to the administration console. See How to Access Oracle Identity Cloud Service if you need help signing in.
2. Click the **Applications** tile. A list of the applications is displayed.
3. Click the application used to generate the token to view its details.
4. In the upper right corner of the page, click **Deactivate**.
5. At the prompt, click **Deactivate Application**.

To re-enable developers to use the same tokens, click **Activate**.

CHAPTER 3 Archive Storage

This chapter explains how to upload, manage, and access data using Archive Storage.

Overview of Archive Storage

Oracle Cloud Infrastructure offers two distinct storage class tiers to address the need for both performant, frequently accessed "hot" storage, and less frequently accessed "cold" storage. Storage tiers help you maximize performance where appropriate and minimize costs where possible.

- Use **Archive Storage** for data to which you seldom or rarely access, but that must be retained and preserved for long periods of time. The cost efficiency of the Archive Storage offsets the long lead time required to access the data.
- Use **Object Storage** for data to which you need fast, immediate, and frequent access. Data accessibility and performance justifies a higher price point to store data in the Object Storage. For more information, see Overview of Object Storage.

About Archive Storage

Archive Storage is ideal for storing data that is accessed infrequently and requires long retention periods. Archive Storage is more cost effective than Object Storage for preserving cold data for:

- Compliance and audit mandates
- Retroactively analyzing log data to determine usage pattern or debug problems
- Historical or infrequently accessed content repository data
- Application generated data that requires archival for future analysis or legal purposes

Unlike Object Storage, Archive Storage data retrieval is **not** instantaneous.

Using Archive Storage



Important

You interact with the data stored in the Archive Storage using the same resources and management interfaces that you use for data stored in Object Storage.

The following summarizes the Object Storage resources you use to store and manage Archive Storage data:

Buckets

Buckets are logical containers for storing objects. A bucket is associated with a single compartment that has policies that determine what actions a user can perform on a bucket and on all the objects in the bucket.

You decide which storage tier (Archive Storage or standard Object Storage) is appropriate for your data when you initially create the bucket container for your data. The storage tier is expressed as a property of the bucket. A bucket's storage tier property sets the initial storage tier of objects added to the bucket. However, objects placed in standard tier buckets can be archived automatically by Object Storage (while remaining in the standard tier bucket) if they meet the criteria of an object lifecycle policy rule in effect for the bucket.

Once set, you cannot change the storage tier property for a bucket:

- An existing Object Storage bucket cannot be downgraded to an Archive Storage bucket.
- An Archive Storage bucket cannot be upgraded to an Object Storage bucket.

In addition to the inability to change the storage tier designation of a bucket, there are other reasons why storage tier selection for buckets requires careful consideration:

- The minimum retention requirement for Archive Storage is 90 days. If you delete objects from Archive Storage before the minimum retention requirements are met, you

are charged a deletion penalty. The deletion penalty is the prorated cost of storing the data for the full 90 days.

- While Archive Storage is more cost effective than Object Storage for cold storage, understand that when you restore objects, you are returning those objects to Object Storage. You are billed for that storage service class while the objects reside in that tier.

You can use object lifecycle policy rules to automatically delete objects in an Archive Storage bucket based on the age of the object.



Important

You *cannot* use object lifecycle policy rules to automatically restore archived objects to the regular Object Storage tier. See Restoring and Downloading Objects for information on restoring objects.

See Managing Buckets for detailed instructions on creating an Archive Storage bucket.

Objects

Any type of data, regardless of content type, is stored as an object. The object is composed of the object itself and metadata about the object. Each object is stored in a bucket.

You upload objects to an Archive Storage bucket the same way you upload objects to a standard Object Storage bucket. The difference is that when you upload an object to an Archive Storage bucket, the object is immediately archived. You must first restore the object before you can download it.

Archived objects are displayed in the object listing of a bucket. You can also display the details of each object.

See Managing Objects for detailed instructions on uploading objects to an Archive Storage bucket.

Restoring and Downloading Objects

To download an object from Archive Storage, you must first restore the object. Restoration takes about four hours from the time an Archive Storage restore request is made, to the time the first byte of data is retrieved. The retrieval time metric is measured by Time To First Byte (TTFB). How long the full restoration takes, depends on the size of the object. You can determine the status of the restoration by looking at the object **Details**. Once the status shows as **Restored**, you can then download the object.

After an object is restored, you have a window of time to download the object. By default, you have 24 hours to download an object, but you can alternatively specify a time from 1 to 240 hours. You can find out how much of the download time is remaining by looking at **Available for Download** in object **Details**. After the allotted download time expires, the object returns to Archive Storage. You always have access to the metadata for an object, regardless of whether the object is in an archived or restored state.

See Managing Objects for detailed instructions on restoring, checking status of, and downloading Archive Storage objects.

Ways to Access Archive Storage

Archive Storage and Object Storage share the *same* management interfaces:

- The Console is an easy-to-use, browser-based interface. To access Archive Storage in the console, do the following:
  - Sign in to the Console.
  - Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**. A list of the buckets in the compartment you're viewing is displayed. If you don’t see the one you're looking for, verify that you’re viewing the correct compartment (select from the list on the left side of the page).
  - Click the name of the Archive Storage tier bucket you want to manage.
- The command line interface (CLI) provides both quick access and full functionality without the need for programming. For more information, see Command Line Interface (CLI).

The syntax for the CLI commands include specifying a service. You will use the Object Storage service designation: `oci os` to manage Archive Storage using the CLI.

- The REST API provides the most functionality, but requires programming expertise. API Reference and Endpoints provides endpoint details and links to the available API reference documents. For general information about using the API, see REST APIs.
- The Oracle Cloud Infrastructure SDKs offer tools to interact with Object Storage and Archive Storage without having to create a framework. For general information about using the SDKs, see Software Development Kits and Command Line Interface.

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

For administrators: Users that need to restore archived objects require the OBJECT_RESTORE permission. See Details for Object Storage, Archive Storage, and Data Transfer for more information.

WORM Compliance

You can achieve WORM compliance with Archive Storage by applying IAM policy permissions so that data once written, cannot be overwritten.

For administrators: There is not a direct way to disallow OBJECT_OVERWRITE. To achieve WORM compliance, you must specifically grant groups OBJECT_CREATE, OBJECT_READ, and OBJECT_INSPECT permissions to keep the data from being overwritten. For example, you can allow groups to inspect objects using a policy like the following:

```
Allow group <group_name> to inspect in compartment <compartment_name>
```

See for more information. If you are new to policies, see Getting Started with Policies and Common Policies.

Limits on Archive Storage Resources

See Service Limits for a list of applicable limits and instructions for requesting a limit increase.

Additional limits include:

- Number of namespaces per root compartment: 1
- Maximum size of object metadata: 2 K

CHAPTER 4 Audit

This chapter explains how to work with audit logs.

Overview of Audit

The Oracle Cloud Infrastructure Audit service automatically records calls to all supported Oracle Cloud Infrastructure public application programming interface (API) endpoints as log events. Currently, all services support logging by Audit. Object Storage service supports logging for bucket-related events, but not for object-related events. Log events recorded by the Audit service include API calls made by the Oracle Cloud Infrastructure Console, Command Line Interface (CLI), Software Development Kits (SDK), your own custom clients, or other Oracle Cloud Infrastructure services. Information in the logs shows what time API activity occurred, the source of the activity, the target of the activity, what the action was, and what the response was.

Each log event includes a header ID, target resources, time stamp of the recorded event, request parameters, and response parameters. You can view events logged by the Audit service by using the Console, API, or the SDK for Java. You can view events, copy the details of individual events, as well as analyze events or store them separately. Data from events can be used to perform diagnostics, track resource usage, monitor compliance, and collect security-related events.

Ways to Access Oracle Cloud Infrastructure

You can access Oracle Cloud Infrastructure using the Console (a browser-based interface) or the REST API. Instructions for the Console and API are included in topics throughout this guide. For a list of available SDKs, see Software Development Kits and Command Line Interface.

To access the Console, you must use a supported browser. Oracle Cloud Infrastructure supports the latest desktop versions of Google Chrome, Microsoft Edge, Internet Explorer 11, Safari, Firefox, and Firefox ESR. Note that private browsing mode is not supported for Firefox, Internet Explorer, or Edge. Mobile browsers are not supported.

For general information about using the API, see REST APIs.

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

Administrators: For an example of policy that gives groups access to audit logs, see Required IAM Policy. To modify the Audit log retention period, you must be a member of the Administrators group. See The Administrators Group and Policy.

Contents of an Audit Log Event

The following explains the contents of a log event. The table does not include request headers that are specific to an Internet browser or other client.

Property	Description
compartmentId	The Oracle Cloud Identifier (OCID) of the compartment.
credentialId	The credential ID of the user. This value is extracted from the HTTP 'Authorization' request header. It consists of the tenantId, userId, and user fingerprint, all delimited by a slash (/).
compartmentName	The name of the compartment. This value is the friendly name associated with compartmentId. This value can change, but Audit logs the value that appeared at the time of the audit event.
eventId	The global unique identifier (GUID) of the event.
eventName	The name of the event.  **Note** Not all services support this property. A null value may appear when not supported by the related service.
eventSource	The source of the event.
eventTime	The time the event occurred, expressed in RFC 3339 timestamp format.
eventType	The type of the event. (Currently, Audit supports only API activities.)

Property	Description
`id`	The OCID of the resource. Found in `responsePayload` along with the name of the resource (`resourceName`). Some API operations generate an audit event, but don't involve a resource, so then the audit event does not have an `id`.
`principalId`	The OCID of the user or service that triggered the event. Find the friendly name of the service or user for this OCID in `userName`.
`requestAction`	The HTTP method of the request.
`requestAgent`	The user agent of the client that made the request.
`requestHeaders`	The HTTP header fields and values in the request.
`requestId`	The opc-request-id of the request. An opc-request-id is a unique Oracle-assigned identifier for the request. If you need to contact Oracle about a particular request, please provide the request ID.
`requestOrigin`	The IP address of the source of the request.
`requestParameters`	The query parameter fields and values for the request.
`responsePayload`	Metadata of interest from the response payload. For example, the OCID of a resource (`id`) and the name of the resource (`resourceName`). Some API operations generate an audit event, but don't involve a resource, so then `responsePayload` is empty.
`requestResource`	The resource targeted by the request.

Property	Description
`resourceName`	The name of the resource. Found in `responsePayload` along with the OCID of the resource (`id`). Some API operations generate an audit event, but don't involve a resource, so then the audit event does not have a `resourceName`.
`responseHeaders`	The headers of the response.
`responseStatus`	The status code of the response.
`responseTime`	The time of the response to the audited request, expressed in RFC 3339 timestamp format.
`tenantId`	The OCID of the tenant.
`userName`	The name of the user or service that triggered this event. This value is the friendly name associated with `principalId`.

Resource Identifiers

Most types of Oracle Cloud Infrastructure resources have a unique, Oracle-assigned identifier called an Oracle Cloud ID (OCID). For information about the OCID format and other ways to identify your resources, see Resource Identifiers.

An Example Log Event

The following is an example log event recorded by the Audit service. Copy the event into a text file to make it easier to read.

```
{
  "requestAgent": "<Browsers, Operating systems>",
  "compartmentName": "CompartmentA",
  "credentialId": "",
```

```
  "responseTime": "2018-06-14T22:24:37.713Z",
  "eventType": "ServiceApi",
  "requestHeaders": {
     <The HTTP header fields and values from the request.>
  },
  "compartmentId": "ocid.compartment.oc1..<compartment_OCID>",
  "requestId": "example-4092-8233-
e8371f94/example1BF1CE734676108C6345FF/51FE3CACE106DD8F825508D04E91E261",
  "eventName": "LaunchInstance",
  "eventSource": "ExampleService",
  "responseStatus": "200",
  "requestParameters": {
    "compartmentId": [
      "example.compartment.region1..<compartment_OCID>"
    ],
    "availabilityDomain": [
      "Example-AD-2"
    ],
    "sortOrder": [
      "DESC"
    ],
    "limit": [
      "25"
    ],
    "sortBy": [
      "timeCreated"
    ]
  },
  "userName": "JohnSmith",
  "responsePayload":{
    "resourceName":"example-instance-name","id":"ocid1.instance.oc1.phx.<instance_OCID>
    },
  "requestAction": "GET",
  "tenantId": "example.tenancy.oc1..<tenancy_OCID>",
  "responseHeaders": {
    "Access-Control-Expose-Headers": [
      "opc-previous-page,opc-next-page,opc-client-info,ETag,opc-request-id,Location"
    ],
    "Access-Control-Allow-Origin": [
      "https://console.us-phoenix-1.oraclecloud.com"
    ],
    "Access-Control-Allow-Credentials": [
```

```
      "true"
    ],
    "Connection": [
      "close"
    ],
    "Content-Length": [
      "3"
    ],
    "opc-request-id": [
      "example-4092-8233-EXAMPLEB863DC6CEC1BF1CE734676108C6345FF/51FE3CACE106DD8F825508D04E91E261"
    ],
    "Date": [
      "Thu, 14 Jun 2018 22:24:37 GMT"
    ],
    "Content-Type": [
      "application/json"
    ]
  },
  "principalId": "example.user.oc1..<principal_OCID>",
  "requestOrigin": "172.24.96.35",
  "eventTime": "2018-06-14T22:24:37.671Z",
  "eventId": "examplea9-f488-4842-96cb-a10f2893b369",
  "requestResource": ""/20160918/images/ocid1.image.region1..<resource_OCID>"
}
```

Viewing Audit Log Events

Audit provides records of API operations performed against supported services as a list of log events. The service logs events at both the tenant and compartment level. By default, audit logs are maintained for 90 days. You can configure audit log retention for up to 365 days.

When viewing events logged by Audit, you might be interested in specific activities that happened in the tenancy or compartment and who was responsible for the activity. You will need to know that the approximate time and date something happened and the compartment in which it happened to display a list of log events that includes the activity in question. List log events by specifying a time range on the 24-hour clock in Greenwich Mean Time (GMT), calculating the offset for your local time zone, as appropriate. New activity is appended to the existing list, usually within 15 minutes of the API call, though processing time can vary.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The following policy statement gives the specified group (Auditors) the ability to view all the Audit event logs in the tenancy:

```
Allow group Auditors to read audit-events in tenancy
```

To give the group access to the Audit event logs in a specific compartment only (Project-A), write a policy like the following:

```
Allow group Auditors to read audit-events in compartment Project-A
```

If you're new to policies, see Getting Started with Policies and Common Policies. For more details about policies for the Audit, see Details for the Audit Service.

Searching and Filtering in the Console

When you navigate to Audit in the Console, a list of results is generated for the current compartment. Audit logs are organized by compartment, so if you are looking for a particular event, you must know which compartment the event occurred in. You can filter the list in all the following ways:

- Date and time
- Request Action Types (operations)
- Keywords

For example, users begin to report that their attempts to log in are failing. You want to use Audit to research the problem. Adjust the date and time to search for corresponding failures during a window of time that starts a little before the events were reported. Look for corresponding failures and similar operations preceding the failures to correlate a reason for the failures.



Note

The service logs events at the time they are processed. There can be a delay between the time an operation occurs and when it is processed.

You can filter results by request actions to zero in on only the events with operations that interest you. For example, say that you only want to know about instances that were deleted during a specific time frame. Select a delete request action filter to see only the events with delete operations.

You can also filter by keywords. Keyword filters are powerful when combined with the values from audit event fields. For example, say that you know the user name of an account and want a list of all activity by that account in a particulate time frame. Do a search using the user name as a keyword filter.

Every audit event contains the same fields, so search for values from those fields. To get a better understanding of what values are available, see Contents of an Audit Log Event.

Using the Console

To search log events

1. Open the navigation menu. Under **Governance and Administration**, go to **Governance** and click **Audit**.
   The list of events that occurred in the current compartment is displayed.
2. Click one of the compartments under **Compartment**.
   Audit organizes logs by compartment, so if you are looking for a particular event, you must know which compartment the event occurred in.
3. Click in the **Start Date** box to choose the start date and time for the range of results you want to see. You can click the arrows on either side of the month to go backward or

forward.

4. (Optional) Specify a time by doing one of the following:
   a. Click **Time** and specify an exact start time in thirty-minute increments.
   b. Type an exact time in the **Start Date** box.
      The service uses a 24-hour clock, so you must provide a number between `0` and `23` for the hour. Also remember to calculate the offset between Greenwich Mean Time (GMT) and your local time.
5. Repeat step 3 and 4 to choose an end date and time.



Note

The age of the results available can be between 90 and 365 days, depending on your tenancy's setting for audit log retention period.

6. (Optional) In **Request Action Types**, specify one or more operations with which to filter results.
   - GET
   - POST
   - PUT
   - PATCH
   - DELETE
7. (Optional) In the **Keywords** box, type the text you want to find and click **Search**.
   Tip: If you want to find log events with a specific status code, include quotes (") around the code to avoid results that have those numbers embedded in a longer string.

The results are updated to include only log events that were processed within the time range and filters you specified. If an event occurred in the recent past, you might have to wait to see it in the list. The service typically requires up to 15 minutes for processing.

If there are more than 100 results for the specified time range, you can click the right arrow next to the page number at the bottom of the page to advance to the next page of log events.



Tip

If you get less than 100 results on the last page of a results list, you might still have more results, which you can access by clicking the right arrow. Audit will prompt you if there are more results.

If you want to view all the key-value pairs in a log event, see To view the details of a log event.

To view the details of a log event

View the details of your event:

- To see only the top-level details, click the down arrow to the right of an event.
- To see lower-level details, click **{ . . . }** to the right of the collapsed parameter.

To copy the details of a log event

The following assumes you have expanded a row in your results.

- To copy an entire event, click the clipboard icon to the right of the `event` parameter.
- To copy a portion of an event, click the clipboard icon to the right of the nested parameter or value you want to copy.

The log event is copied to your clipboard. The Audit service logs events in JSON format. You can paste the log event details into a text editor to save and review later or to use with standard log analysis tools.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operation to list audit log events:

- ListEvents

> **Note**
>
> This is a query API. Do not use this API for bulk-export operations.

Setting Audit Log Retention Period

By default, Audit logs are retained for 90 days. You can configure log retention for up to 365 days. You can edit the log retention period in the tenancy details page.

Retention period is a tenancy-level setting. The value of the retention period setting affects all regions and all compartments. You can't set different retention periods for different regions or compartments.

Required IAM Policy

To modify the Audit log retention period, you must be a member of the Administrators group. See The Administrators Group and Policy

Using the Console

To modify the Audit log retention period

1. Open the **User** menu () and click **Tenancy:** *<your_tenancy_name>*.

The tenancy details are displayed. The **Audit Retention Period** is displayed under **Tenancy Information**.

2. Click **Edit Audit Retention Policy**. Enter the number of days you want to retain the audit logs for. The minimum you can set the value to is 90 and the maximum is 365. This value is enforced for all regions and all compartments in the tenancy.
3. Click **Submit**.



Note

You won't see the new value immediately

It may take several minutes for the new setting to display in the Console.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operations to manage the log retention period configuration:

- GetConfiguration
- UpdateConfiguration

Bulk Export of Audit Log Events

You can request a bulk export of audit logs, and within 3–4 business days Oracle support will begin making copies of the logs and adding them to buckets in your tenancy. The export includes logs for the specified regions, beginning after you make the request and continuing into the future.

Highlights

- Administrators have full control of the buckets and can provide access to others with IAM policy statements.
- Exported logs remain available indefinitely.



Tip

You can automatically manage archiving and deleting logs using Object Storage. See Using Object Lifecycle Management.

- Specify all the regions you want exported in your request. If you only request some regions, then decide later you want to add other regions, you must make another request.
- To disable your bulk export, contact Oracle support. New logs will stop being added to the bucket, and audit logs will only be available through the Console, based on the retention period you have defined.

Required IAM Policy

To access the bucket where Oracle exports the audit logs, you must be a member of the Administrators group. See The Administrators Group and Policy

Requesting an Export of Audit Logs

A member of the Administrators group for your tenancy must create a ticket at My Oracle Support and provide the following information:

- Ticket name: Export Audit Logs - *<your_company_name>*
- Tenancy OCID

- Regions

For example:

- Ticket name: Export Audit Logs - ACME
- Tenancy OCID: ocid1.tenancy.oc1.*<unique_ID>*
- Regions: us-ashburn-1, us-phoenix-1



Note

It can take 3–4 business days before your My Oracle Support ticket is complete and the logs are available to you.

Bucket and Object Details

This section specifies the naming conventions of the bucket and objects you receive.

Bucket Name Format

Oracle support creates buckets for audit log exports using the following naming format:

`oci-logs._audit.`*<compartment_OCID>*

- `oci-logs` identifies that Oracle created this bucket.
- `_audit` identifies that the bucket contains audit events.
- *<compartment_OCID>* identifies the compartment where the audit events were generated.

For example:

```
oci-logs._audit.ocid1compartment.oc1..<unique_ID>
```



Important

If the OCID of the compartment that generated the audit log contains a colon, your bucket name will not match the OCID. To create a bucket, Oracle must substitute colon characters (:) from the OCID with dot characters (.) in the bucket name.

Object Name Format

Objects use the following naming format:

<region>/*<ad>*/*<YYYY-MM-DDTHH:MMZ>*[_*<seqNum>*].log.gz

- *<region>* identifies the region where the audit events were generated.
- *<ad>* identifies the availability domain where the audit events were generated.
- *<YYYY-MM-DDTHH:MMZ>* specifies the start of a 10-minute window for audit events listed in the object. For example, during a thirty-minute period, Oracle would write three objects, the first would contain audit events that occurred from 00–09:59, the second object would contain events from 10–19:59, while the third object contains events for 20–29:59.
- [_*<seqNum>*] specifies a conditional sequence number. If present, this number means that either an event came in late or the object became too large to write. Sequence numbers start at two. Apply multiple sequence numbers to the original object in the order listed.

For example:

```
us-phoenix-1/ad1/2019-03-21T00:00Z.log.gz
us-phoenix-1/ad1/2019-03-21T00:00Z_2.log.gz
```

File Format

Files list a single audit event per line. For more information, see Contents of an Audit Log Event.

CHAPTER 5 Block Volume

This chapter explains how to create storage volumes and attach them to instances.

Overview of Block Volume

The Oracle Cloud Infrastructure Block Volume service lets you dynamically provision and manage block storage volumes. You can create, attach, connect, and move volumes as needed to meet your storage and application requirements. After you attach and connect a volume to an instance, you can use the volume like a regular hard drive. You can also disconnect a volume and attach it to another instance without the loss of data.

These components are required to create a volume and attach it to an instance:

- **Instance:** A bare metal or virtual machine (VM) host running in the cloud.
- **Volume attachment:** There are two types of volume attachments:
  - iSCSI**:** A TCP/IP-based standard used for communication between a volume and attached instance.
  - Paravirtualized**:** A virtualized attachment available for VMs.
- **Volume:** There are two types of volumes:
  - **Block volume:** A detachable block storage device that allows you to dynamically expand the storage capacity of an instance.
  - **Boot volume:** A detachable boot volume device that contains the image used to boot a Compute instance. See Boot Volumes for more information.

For additional Oracle Cloud Infrastructure terms, see the Glossary.

Typical Block Volume Scenarios

Scenario A: Expanding an Instance's Storage

A common usage of Block Volume is adding storage capacity to an Oracle Cloud Infrastructure instance. After you have launched an instance and set up your cloud network, you can create a block storage volume through the Console or API. Then, you attach the volume to an instance using a volume attachment. After the volume is attached, you connect to the volume from your instance's guest OS using iSCSI. The volume can then be mounted and used by your instance.

Scenario B: Persistent and Durable Storage

A Block Volume volume can be detached from an instance and moved to a different instance without the loss of data. This data persistence enables you to migrate data between instances and ensures that your data is safely stored, even when it is not connected to an instance. Any data remains intact until you reformat or delete the volume.

To move your volume to another instance, unmount the drive from the initial instance, terminate the iSCSI connection, and attach the volume to the second instance. From there, you connect and mount the drive from that instance's guest OS to have access to all of your data.

Additionally, Block Volume volumes offer a high level of data durability compared to standard, attached drives. All volumes are automatically replicated for you, helping to protect against data loss.

Scenario C: Instance Scaling

When you terminate an instance, you can keep the associated boot volume and use it to launch a new instance with a different instance type or shape. This allows you to easily switch from a bare metal instance to a VM instance and vice versa, or scale up or scale down the number of cores for an instance. See Creating an Instance for steps to launch an instance based on a boot volume.

Volume Attachment Types

When you attach a block volume to a VM instance, you have two options for attachment type, iSCSI or paravirtualized. Paravirtualized attachments simplify the process of configuring your block storage by removing the extra commands that are required before connecting to an iSCSI-attached volume. The trade-off is that IOPS performance for iSCSI attachments is greater than that for paravirtualized attachments. You should consider your requirements when selecting a volume's attachment type.



Important

Connecting to Volumes on Linux Instances

When connecting to volumes on Linux instances, if you want to automatically mount these volumes on instance boot, you need to use some specific options in the `/etc/fstab` file, or the instance may fail to launch. See Traditional fstab Options and fstab Options for Block Volumes Using Consistent Device Paths for more information.

iSCSI

iSCSI attachments are the only option when connecting a block volume to any of the following types of instances:

- Bare metal instances
- VM instances based on Windows images that were published before February 2018
- VM instances based on Linux images that were published before December 2017

After the volume is attached, you need to log in to the instance and use the `iscsiadm` command-line tool to configure the iSCSI connection. For more information about the additional configuration steps required for iSCSI attachments, see iSCSI Commands and Information, Connecting to a Volume, and Disconnecting From a Volume.

IOPS performance is better with iSCSI attachments compared to paravirtualized attachments. For more information about iSCSI-attached volume performance, see Block Volume Performance.

Paravirtualized

Paravirtualized attachments are an option when attaching volumes to the following types of VM instances:

- For VM instances launched from Oracle-provided images, you can select this option for Linux-based images published in December 2017 or later, and Windows images published in February 2018 or later.
- For VM instances launched from custom images, the volume attachment type is based on the volume attachment type from the VM the custom image was created from.

After you attach a volume using the paravirtualized attachment type, it is ready to use, and you do not need to run any additional commands. However, because of the overhead of virtualization, this reduces the maximum IOPS performance for larger block volumes. See Paravirtualized Attachment Performance for more information.

Volume Access Types

When you attach a block volume, you can specify one of the following options for access type:

- **Read/write:** This is the default option for volume attachments. With this option, an instance can read and write data to the volume.
- **Read-only:** With this option, an instance can only read data on the volume. It cannot update data on the volume. Specify this option to safeguard data against accidental or malicious modifications.

To change the access type for a block volume, you need to detach the volume and specify the new access type when you reattach the volume. For more information, see Detaching a Volume and Attaching a Volume.

The access type for boot volumes is always read/write. If you want to change the access type, you need to stop the instance and detach the boot volume. You can then reattach it to another

instance as a block volume, with read-only specified as the access type. For more information, see Detaching a Boot Volume and Attaching a Volume.

Device Paths

When you attach a block volume to a compatible Linux-based instance, you can select a device path that remains consistent between instance reboots. This enables you to refer to the volume using a consistent device path. For example, you can use the device path when you set options in the `/etc/fstab` file to automatically mount the volume on instance boot.

Consistent device paths are supported on instances when all of the following things are true:

- The instance was created using an Oracle-provided image.
- The image is a Linux-based image.
- The image was released in November 2018 or later. For specific version numbers, see Oracle-Provided Image Release Notes.
- The instance was launched after January 11, 2019.

For instances launched using the image OCID or an existing boot volume, if the source image supports consistent device paths, the instance supports device paths.

Consistent device paths are not supported on Linux-based partner images or custom images that are created from other sources. This feature does not apply to Windows-based images.



Important

You must select a device path when you attach a volume using the Console, it is required. Specifying a device path is optional when you attach a volume using the CLI, REST APIs, or SDK.

For more information about consistent device paths, see Connecting to Volumes With Consistent Device Paths.

Regions and Availability Domains

Volumes are only accessible to instances in the same availability domain . You cannot move a volume between availability domains or regions.

For more information, see Regions and Availability Domains.

Resource Identifiers

Most types of Oracle Cloud Infrastructure resources have a unique, Oracle-assigned identifier called an Oracle Cloud ID (OCID). For information about the OCID format and other ways to identify your resources, see Resource Identifiers.

Ways to Access Oracle Cloud Infrastructure

You can access Oracle Cloud Infrastructure using the Console (a browser-based interface) or the REST API. Instructions for the Console and API are included in topics throughout this guide. For a list of available SDKs, see Software Development Kits and Command Line Interface.

To access the Console, you must use a supported browser. Oracle Cloud Infrastructure supports the latest desktop versions of Google Chrome, Microsoft Edge, Internet Explorer 11, Safari, Firefox, and Firefox ESR. Note that private browsing mode is not supported for Firefox, Internet Explorer, or Edge. Mobile browsers are not supported.

For general information about using the API, see REST APIs.

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see

Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

Monitoring Resources

You can monitor the health, capacity, and performance of your Oracle Cloud Infrastructure resources by using metrics, alarms, and notifications. For more information, see Monitoring Overview and Notifications Overview.

Tagging Resources

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

Block Volume Encryption

The Oracle Cloud Infrastructure Block Volume service always encrypts all block volumes, boot volumes, and volume backups at rest by using the Advanced Encryption Standard (AES) algorithm with 256-bit encryption. By default all volumes and their backups are encrypted using the Oracle-provided encryption keys. You have the option to encrypt all of your volumes and their backups using the keys that you own and manage using the Key Management service, for more information see Overview of Key Management. If you do not configure a volume to use the Key Management service or you later unassign a key from the volume, the Block Volume service uses the Oracle-provided encryption key instead. This applies to both encryption at-rest and in-transit encryption.

For how to use your own key for new volumes, see Creating a Volume. See To assign a key to an existing Block Volume for how to assign or change the key for an existing volume.

All the data moving between the instance and the block volume is transferred over an internal and highly secure network. If you have specific compliance requirements related to the encryption of the data while it is moving between the instance and the block volume, the Block Volume service provides the option to enable in-transit encryption for paravirtualized volume attachments on virtual machine (VM) instances.



Important

In-transit encryption for boot and block volumes is only available for virtual machine (VM) instances launched from Oracle-provided images, it is not supported on bare metal instances. It is also not supported in most cases for instances launched from custom images imported for "bring your own image" (BYOI) scenarios. To confirm support for certain Linux-based custom images and for more information contact Oracle support, see Contacting Support.

Block Volume Data Eradication

The Oracle Cloud Infrastructure Block Volume service uses eventual-overwrite data eradication, which guarantees that block volumes you delete cannot be accessed by anyone else and that the deleted data is eventually overwritten. When you terminate a volume, its associated data is overwritten in the storage infrastructure before any future volume allocations.

Block Volume Capabilities and Limits

Block Volume volumes can be created in sizes ranging from **50 GB** to **32 TB** in **1 GB** increments. By default, Block Volume volumes are **1 TB**.

Block Volume volume performance varies with volume size.

See Service Limits for a list of applicable limits and instructions for requesting a limit increase.

Additional limits include:

- **Volumes per instance:** 32
- **Number of backups**
  - Monthly universal credits: 1000
  - Pay-as-you-go: 500

iSCSI Commands and Information

Block volumes attached with the iSCSI attachment type use the iSCSI protocol to connect a volume to an instance. See Volume Attachment Types for more information about volume attachment options.

Once the volume is attached, you need to log on to the instance and use the `iscsiadm` command-line tool to configure the iSCSI connection. After you configure the volume, you can mount it and use it like a normal hard drive.

To enhance security, Oracle enforces an iSCSI security protocol called CHAP that provides authentication between the instance and volume.

Accessing a Volume's iSCSI Information

When you successfully attach a volume to an instance, Block Volume provides a list of iSCSI information. You need the following information from the list when you connect the instance to the volume.

- IP address
- Port
- CHAP user name and password (if enabled)
- IQN



Note

The CHAP credentials are auto-generated by the system and cannot be changed. They are also unique to their assigned volume/instance pair and cannot be used to authenticated another volume/instance pair.

The Console provides this information on the details page of the volume's attached instance. Click the Actions icon (three dots) on your volume's row, and then click **iSCSI Information**. The system also returns this information when the `AttachVolume` API operation completes successfully. You can re-run the operation with the same parameter values to review the information.

See Attaching a Volume and Connecting to a Volume for step-by-step instructions.

Additional Reading

There is a wealth of information on the internet about iSCSI and CHAP. If you need more information on these topics, try the following pages:

- Oracle Linux Administrator's Guide for Release 7 - About iSCSI Storage
- Oracle Linux Administrator's Guide for Release 6 - About iSCSI Storage
- Troubleshooting iSCSI Configuration Problems

Volume Groups

The Oracle Cloud Infrastructure Block Volume service provides you with the capability to group together multiple volumes in a volume group. A volume group can include both types of volumes, boot volumes, which are the system disks for your Compute instances, and block volumes for your data storage. You can use volume groups to create volume group backups and clones that are point-in-time and crash-consistent.

This simplifies the process to create time-consistent backups of running enterprise applications that span multiple storage volumes across multiple instances. You can then restore an entire group of volumes from a volume group backup.

Similarly, you can also clone an entire volume group in a time-consistent and crash-consistent manner. A deep disk-to-disk and fully isolated clone of a volume group, with all the volumes associated in it, becomes available for use within a matter of seconds. This speeds up the process of creating new environments for development, quality assurance, user acceptance testing, and troubleshooting.

For more information about Block Volume-backed system disks, see Boot Volumes. For more information about Block Volume backups see Overview of Block Volume Backups. See Cloning a Volume for more information about Block Volume clones.

This capability is available using the Console, command line interface (CLI), SDKs, or REST APIs.

Volume groups and volume group backups are high-level constructs that allow you to group together multiple volumes. When working with volume groups and volume group backups, keep the following in mind:

- You can only add a volume to a volume group when the volume status is available.
- You can add up to 32 volumes in a volume group, up to a maximum size limit of 128 TB. For example, if you wanted to add 32 volumes of equal size to a volume group, the maximum size for each volume would be 4 TB. Or you could add volumes that vary in size, however the overall combined size of all the block and boot volumes in the volume group must be 128 TB or less. Make sure you account for the size of any boot volumes in your volume group when considering volume group size limits.
- Each volume may only be in one volume group.
- When you clone a volume group, a new group with new volumes are created. For example, if you clone a volume group containing three volumes, once this operation is complete, you will now have two separate volume groups and six different volumes with nothing shared between the volume groups.
- When you update a volume group using the CLI, SDKs, or REST APIs you need to specify all the volumes to include in the volume group each time you use the update operation.

If you do not include a volume ID in the update call, that volume will be removed from the volume group.

- When you delete a volume group the individual volumes in the group are not deleted, only the volume group is deleted.
- When you delete a volume that is part of a volume group you must first remove it from the volume group before you can delete it.
- When you delete a volume group backup, all the volume backups in the volume group backup are deleted.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let volume admins manage block volumes, backups, and volume groups lets the specified group do everything with block volumes, backups, and volume groups.

See the following policy examples for working with volume groups:

- Let users create a volume group lets the specified group create a volume group from a set of volumes.
- Let users clone a volume group lets the specified group clone a volume group from an existing volume group.
- Let users create a volume group backup lets the specified group create a volume group backup.
- Let users restore a volume group backup lets the specified group create a volume group by restoring a volume group backup.



Tip

When users create a backup from a volume or restore a volume from a backup, the volume and backup don't have to be in the same compartment. However, users must have access to both compartments.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Tagging Resources

You can apply tags to your resources to help you organize them according to your business needs. You can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

Using the Console

To create a volume group from existing volumes

1. Open the navigation menu. Under **Core Infrastructure**, go to **Block Storage** and click **Volumes Groups**.
2. Click **Create Volume Group**.
3. Fill in the required volume information:
    - **Name**: A user-friendly name or description.
    - **Compartment**: The compartment for the volume group.
    - **Availability Domain**: The availability domain for the volume group.

- **Volumes**: For each volume you want to add, select the compartment containing the volume and then the volume to add. Click **+ Volume** to add additional volumes.

4. Click **Create Volume Group**.

To view the volumes in a volume group

1. Open the navigation menu. Under **Core Infrastructure**, go to **Block Storage** and click **Volumes Groups**.
2. In the **Volume Groups** list, click the volume group you want to view the volumes for.
3. To view the block volumes for the volume group, in **Resources**, click **Block Volumes**.
4. To view the boot volumes for the volume group, in **Resources**, click **Boot Volumes**.

To add block volumes to an existing volume group

1. Open the navigation menu. Under **Core Infrastructure**, go to **Block Storage** and click **Volumes Groups**.
2. In the **Volume Groups** list, click the volume group you want to add the volume to.
3. In **Resources**, click **Block Volumes**.
4. Click **Add Block Volumes**.
5. For each block volume you want to add, select the compartment containing the volume and then select the volume to add. Click **+ Volume** to add additional volumes.
6. Once you have selected all the block volumes to add to the volume group, click **Add**.

To remove block volumes from an existing volume group

1. Open the navigation menu. Under **Core Infrastructure**, go to **Block Storage** and click **Volumes Groups**.
2. In the **Volume Groups** list, click the volume group you want to add the volume to.

3. In **Resources**, click **Block Volumes**.
4. In **Actions** menu for the block volume you want to remove, click **Remove**.
5. In the **Confirm** dialog, click **Remove**.

To add boot volumes to an existing volume group

1. Open the navigation menu. Under **Core Infrastructure**, go to **Block Storage** and click **Volumes Groups**.
2. In the **Volume Groups** list, click the volume group you want to add the volume to.
3. In **Resources**, click **Boot Volumes**.
4. Click **Add Boot Volumes**.
5. For each boot volume you want to add, select the compartment containing the volume and then select the volume to add. Click **+ Volume** to add additional volumes.
6. Once you have selected all the boot volumes to add to the volume group, click **Add**.

To remove boot volumes from an existing volume group

1. Open the navigation menu. Under **Core Infrastructure**, go to **Block Storage** and click **Volumes Groups**.
2. In the **Volume Groups** list, click the volume group you want to add the volume to.
3. In **Resources**, click **Boot Volumes**.
4. In **Actions** menu for the boot volume you want to remove, click **Remove**.
5. In the **Confirm** dialog, click **Remove**.

To create a backup of the volume group

1. Open the navigation menu. Under **Core Infrastructure**, go to **Block Storage** and click **Volumes Groups**.

2. In the **Volume Groups** list, click **Create Volume Group Backup** in the **Actions** menu for the volume group you want to create a backup for.

To create a clone of the volume group

1. Open the navigation menu. Under **Core Infrastructure**, go to **Block Storage** and click **Volumes Groups**.
2. In the **Volume Groups** list, click **Create Volume Group Clone** in the **Actions** menu for the volume group you want to clone.

To delete the volume group

1. Open the navigation menu. Under **Core Infrastructure**, go to **Block Storage** and click **Volumes Groups**.
2. In the **Volume Groups** list, click the volume group you want to delete.
3. On the **Volume Group Details** page, click **Terminate**.
4. On the **Terminate Volume Group** dialog, click **Terminate**.



Note

When you delete a volume group the individual volumes in the group are not deleted, only the volume group is deleted.

To restore a volume group from a volume group backup

1. Open the navigation menu. Under **Core Infrastructure**, go to **Block Storage** and click **Volumes Group Backups**.

2. In the **Volume Group Backups** list, click the volume group backup you want to restore.
3. Click **Create Volume Group**.
4. Fill in the required volume information:
   - **Name**: A user-friendly name or description.
   - **Compartment**: The compartment for the volume group.
   - **Availability Domain**: The availability domain for the volume group.
5. Click **Create Volume Group**.

Using the CLI

For information about using the CLI, see Command Line Interface (CLI).

To retrieve information about the supported operations

Open a command prompt and run the one of the following commands to retrieve the information.

- To retrieve the supported operations for volume groups:

```
oci bv volume-group --help
```

- To retrieve the supported operations for volume group backups:

```
oci bv volume-group-backup --help
```

- To retrieve help for a specific volume group operation:

```
oci bv volume-group <operation_name> --help
```

- To retrieve help for a specific volume group backup operation:

```
oci bv volume-group-backup <operation_name> --help
```

Volume Group Operations

To list the volume groups in a specified compartment

Open a command prompt and run:

```
oci bv volume-group list --compartment-id <compartment_ID>
```

For example:

```
oci bv volume-group list --compartment-id ocid1.compartment.oc1..<unique_ID>
```

To create a volume group from existing volumes

Open a command prompt and run:

```
oci bv volume-group create --compartment-id <compartment_ID> --availability-domain <external_AD> --
source-details <Source_details_JSON>
```

Volume status must be available to add it to a volume group.

For example:

```
oci bv volume-group create --compartment-id ocid1.compartment.oc1..<unique_ID> --availability-domain
ABbv:PHX-AD-1 --source-details '{"type": "volumeIds", "volumeIds":["ocid1.volume.oc1.phx.<unique_ID_1>",
"ocid1.volume.oc1.phx.<unique_ID_2>"]}'
```

To clone a volume group from another volume group

Open a command prompt and run:

```
oci bv volume-group create --compartment-id <compartment_ID> --availability-domain <external_AD> --
source-details <Source_details_JSON>
```

For example:

```
oci bv volume-group create --compartment-id ocid1.compartment.oc1..<unique_ID> --availability-domain
ABbv:PHX-AD-1 --source-details '{"type": "volumeGroupId", "volumeGroupId":
"ocid1.volumegroup.oc1.phx.<unique_ID>"}'
```

To restore a volume group from a volume group backup

Open a command prompt and run:

```
oci bv volume-group create --compartment-id <compartment_ID> --availability-domain <external_AD> --
source-details <Source_details_JSON>
```

For example:

```
oci bv volume-group create --compartment-id ocid1.compartment.oc1..<unique_ID> --availability-domain
ABbv:PHX-AD-1 --source-details '{"type": "volumeGroupBackupId", "volumeGroupBackupId":
"ocid1.volumegroup.oc1.sea.<unique_ID>"}'
```

To retrieve a volume group

Open a command prompt and run:

```
oci bv volume-group get --volume-group-id <volume-group-ID>
```

For example:

```
oci bv volume-group get --volume-group-id ocid1.volumegroup.oc1.phx.<unique_ID>
```

To update display name or add/remove volumes from a volume group

Open a command prompt and run:

```
oci bv volume-group update --volume-group-id <volume-group_ID> --volume-ids <volume_ID_JSON>
```

You can update the volume group display name along with adding or removing volumes from the volume group. The volume group is updated to include only the volumes specified in the update operation. This means that you need to specify the volume IDs for all of the volumes in the volume group each time you update the volume group.

The following example changes the volume group's display name for a volume group with two volumes:

```
oci bv volume-group update --volume-group-id ocid1.volumegroup.oc1.phx.<unique_ID> --volume-ids '
["ocid1.volume.oc1.phx.<unique_ID_1>","ocid1.volume.oc1.phx.<unique_ID_2>"]' --display-name "new display
name"
```

If you specify volumes in the command that are not part of the volume group they are added to the group. Any volumes not specified in the command are removed from the volume group.

To delete a volume group

Open a command prompt and run:

```
oci bv volume-group delete --volume-group-id <volume-group_ID>
```

When you delete a volume group, the individual volumes in the group are not deleted, only the volume group is deleted.

For example:

```
oci bv volume-group delete --volume-group-id ocid1.volumegroup.oc1.phx.<unique_ID>
```

Volume Group Backup Operations

To list volume backup groups

Open a command prompt and run:

```
oci bv volume-group-backup list --compartment-id <compartment_ID>
```

For example:

```
oci bv volume-group-backup list --compartment-id ocid1.compartment.oc1..<unique_ID>
```

To create a volume group backup

Open a command prompt and run:

```
oci bv volume-group-backup create --volume-group-id <volume-group_ID>
```

For example:

```
oci bv volume-group-backup create --volume-group-id ocid1.volumegroup.oc1.phx.<unique_ID>
```

To retrieve a volume group backup

Open a command prompt and run:

```
oci bv volume-group-backup get --volume-group-backup-id <volume-group-backup_ID>
```

For example:

```
oci bv volume-group-backup get --volume-group-backup-id ocid1.volumegroupbackup.oc1.phx.<unique_ID>
```

To update display name for a volume group backup

Open a command prompt and run:

```
oci bv volume-group-backup update --volume-group-backup-id <volume-group-backup_ID> --display-name <new_display_name>
```

You can only update the display name for the volume group backup.

For example:

```
oci bv volume-group-backup update --volume-group-backup-id ocid1.volumegroupbackup.oc1.phx.<unique_ID> --display-name "new display name"
```

To delete a volume group backup

Open a command prompt and run:

```
oci bv volume-group-backup delete --volume-group-backup-id <volume-group-backup_ID>
```

When you delete a volume group backup, all volume backups in the group are deleted.

For example:

```
oci bv volume-group-backup delete --volume-group-backup-id ocid1.volumegroupbackup.oc1.phx.<unique_ID>
```

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operations for working with volume groups:

- ListVolumeGroups
- CreateVolumeGroup
- DeleteVolumeGroup
- GetVolumeGroup
- UpdateVolumeGroup

Use the following operations for working with volume group backups:

- ListVolumeGroupBackups
- CreateVolumeGroupBackup
- DeleteVolumeGroupBackup
- GetVolumeGroupBackup
- UpdateVolumeGroupBackup

Creating a Volume

You can create a volume using Block Volume.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let volume admins manage block volumes, backups, and volume groups lets the specified group do everything with block volumes and backups.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Monitoring Resources

You can monitor the health, capacity, and performance of your Oracle Cloud Infrastructure resources by using metrics, alarms, and notifications. For more information, see Monitoring Overview and Notifications Overview.

Tagging Resources

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

Using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Block Storage** and click **Block Volumes**.
2. Click **Create Block Volume**.
3. Fill in the required volume information:
    - **Name**: A user-friendly name or description.
    - **Domain**: Must be in the same **availability domain** as the instance.
    - **Size**: Must be between **50 GB** and **32 TB**. You can choose in 1 GB increments within this range. The default is 1024 GB. If you choose a size outside of your service limit, you may be prompted to request an increase. For more information, see Service Limits.
    - **Backup Policy**: Optionally, you can select the appropriate backup policy for your requirements. See Policy-Based Backups for more information about backup policies.
    - **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you

should apply tags, skip this option (you can apply tags later) or ask your administrator.

- **Encryption:** Optionally, you can encrypt the data in this volume using your own Key Management encryption key. To use Key Management for your encryption needs, select the **Encrypt using Key Management** check box. Then, select the **Vault Compartment** and **Vault** that contain the master encryption key you want to use. Also select the **Master Encryption Key Compartment** and **Master Encryption Key**. For more information about encryption, see Overview of Key Management.

4. Click **Create Block Volume**.
   The volume will be ready to attach once its icon no longer lists it as **PROVISIONING** in the volume list. For more information, see Attaching a Volume.

Using the API

To create a volume, use the following operation:

- CreateVolume

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Attaching a Volume

You can attach a volume to an instance in order to expand the available storage on the instance. If you specify iSCSI as the volume attachment type, you must also connect and mount the volume from the instance for the volume to be usable. For more information, see Volume Attachment Types and Connecting to a Volume.



Note

You should only attach Linux volumes to Linux instances and Windows volumes to Windows instances.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let users launch Compute instances includes the ability to attach/detach existing block volumes. The policy in Let volume admins manage block volumes, backups, and volume groups lets the specified group do everything with block volumes and backups, but not launch instances.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
2. In the **Instances** list, click the instance that you want to attach a volume to.
3. In the **Resources** section, click **Attached Block Volumes**.
4. Click **Attach Block Volume**.
5. Select the volume attachment type, **iSCSI** or **Paravirtualized**.
   For more information, see Volume Attachment Types.

6. In the **Block Volume Compartment** drop-down list, select the compartment.
7. Specify the volume you want to attach to. To use the volume name, choose **SELECT VOLUME** and then select the volume from the **Block Volume** drop-down list. To specify the volume OCID, choose **ENTER VOLUME OCID** and then enter the OCID into the **Block Volume OCID** field.
8. If the instance supports consistent device paths, and the volume you are attaching is not a boot volume, select a path from the **Device Path** drop-down list when attaching. This is required and enables you to specify a device path for the volume attachment that remains consistent between instance reboots.

   For more information about this feature and the instances that support it, see Connecting to Volumes With Consistent Device Paths

   Tip

   You must select a device path when you attach a volume from the Console, it is not optional. Specifying a device path is optional when you attach a volume using the CLI, REST APIs, or SDK.
9. Select the access type, **Read/Write** or **Read-only**.

   For more information, see Volume Access Types.
10. For paravirtualized volume attachments on virtual machine (VM) instances, you can optionally encrypt data that is transferred between the instance and the Block Volume service storage servers. To do this, select the **Use in-transit encryption** check box. If you configured the volume to use an encryption key that you manage using the Key Management service, this key is used for in-transit encryption. Otherwise, the Oracle-provided encryption key is used. See Block Volume Encryption for more information.
11. Click **Attach**.

    When the volume's icon no longer lists it as **Attaching**, if the attachment type is Paravirtualized, you can use the volume. If the attachment type is iSCSI, you need to connect to the volume first. For more information, see Connecting to a Volume.

On Linux-based instances, if you want to automatically mount volumes on instance boot, you need to set some specific options in the `/etc/fstab` file, or the instance may fail to launch. This applies to both iSCSI and paravirtualized attachment types. For volumes using consistent device paths, see fstab Options for Block Volumes Using Consistent Device Paths. For all other volumes, see Traditional fstab Options.

Using the API

To attach a volume to an instance, use the following operation:

- AttachVolume

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Connecting to Volumes With Consistent Device Paths

Oracle Cloud Infrastructure supports consistent device paths for block volumes that are attached to compatible Linux-based instances. When you attach a block volume to an instance, you must select a device path that remains consistent between instance reboots. This enables you to use a consistent device path when you refer to the volume to perform tasks such as:

- Creating partitions.
- Creating file systems.
- Mounting file systems.
- Specifying options in the `/etc/fstab` file to ensure that volumes are mounted properly when automatically mounting volumes on instance boot. For more information, see fstab Options for Block Volumes Using Consistent Device Paths.

When you use consistent device paths on compatible Linux-based instances, the boot volume's device path is:

```
/dev/oracleoci/oraclevda
```



Note

Device paths are not available when you attach a boot volume as a data volume to a second instance.

Images that Support Consistent Device Paths

Consistent device paths are supported on instances when all of the following things are true:

- The instance was created using an Oracle-provided image.
- The image is a Linux-based image.
- The image was released in November 2018 or later. For specific version numbers, see Oracle-Provided Image Release Notes.
- The instance was launched after January 11, 2019.

For instances launched using the image OCID or an existing boot volume, if the source image supports consistent device paths, the instance supports device paths.

Consistent device paths are not supported on Linux-based partner images or custom images that are created from other sources. This feature does not apply to Windows-based images.



Important

You must select a device path when you attach a volume using the Console, it is required. Specifying a device path is optional when you attach a volume using the CLI, REST APIs, or SDK.

Device Paths in the Console

You select a device path when you attach a block volume to an instance.

If you specify a device path, the path appears in the **Attached Block Volumes** list for an instance, in the **Device Path** field. An example is shown in the following screenshot.



Device Paths on the Instance

Use the following sample commands to perform various configuration tasks on the attached volume. Commands are provided for volumes that use consistent device paths and for volumes that don't.

Creating a partition with `fdisk`

- **No device path specified:**

```
fdisk /dev/sdb
```

- **Device path specified:**

```
fdisk /dev/oracleoci/oraclevdb
```

Creating an ext3 file system

- **No device path specified:**

```
/sbin/mkfs.ext3 /dev/sdb1
```

- **Device path specified:**

```
/sbin/mkfs.ext3 /dev/oracleoci/oraclevdb1
```

Updating the `/etc/fstab` file

- **No device path specified:**

```
UUID=84dc162c-43dc-429c-9ac1-b511f3f0e23c /oradiskvdb1 xfs defaults,_netdev,noatime 0 2
```

- **Device path specified:**

```
/dev/oracleoci/oraclevdb1   /oradiskvdb1    ext3    defaults,_netdev,noatime  0  2
```

Mounting the file system

- **No device path specified:**

```
mount /dev/sdb1 /oradiskvdb1
```

- **Device path specified:**

```
mount /dev/oracleoci/oraclevdb1 /oradiskvdb1
```

Connecting to a Volume

For volumes attached with Paravirtualized as the volume attachment type, you do not need to perform any additional steps after Attaching a Volume, the volumes are connected automatically. However, for Linux-based images, if you want to mount these volumes on instance boot, you need to perform additional configuration steps. If you specified a device path when you attached the volume, see fstab Options for Block Volumes Using Consistent Device Paths. If you did not specify a device path or if your instance was created from an image that does not support device paths, see Traditional fstab Options.

For volumes attached with iSCSI as the volume attachment type, you need to connect and mount the volume from the instance for the volume to be usable. For more information about attachment type options, see Volume Attachment Types. In order to connect the volume, you must first attach the volume to the instance, see Attaching a Volume.

Connecting to iSCSI-Attached Volumes

Required IAM Policy

Connecting a volume to an instance does not require a specific IAM policy. However, you may need permission to run the necessary commands on the instance's guest OS. Contact your system administrator for more information.

Prerequisites

You must attach the volume to the instance before you can connect the volume to the instance's guest OS. For details, see Attaching a Volume.

To connect the volume, you need the following information:

- iSCSI IP Address
- iSCSI Port numbers
- CHAP credentials (if you enabled CHAP)
- IQN

The Console provides the commands required to configure, authenticate, and log on to iSCSI.

Connecting to a Volume on a Linux Instance

1. Use the Console to obtain the iSCSI data you need to connect the volume:
    a. Log on to Oracle Cloud Infrastructure.
    b. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
    c. Click the name of the instance to display the instance details.
    d. In the **Resources** section on the **Instance Details** page, click **Attached Block Volumes** to view the attached block volume.

e. Click the Actions icon (three dots) next to the volume you're interested in, and then click **iSCSI Commands and Information**.

   The **iSCSI Commands and Information** dialog box displays specific identifying information about your volume and the iSCSI commands you'll need. The commands are ready to use with the appropriate information included. You can copy and paste the commands into your instance session window for each of the following steps.

2. Log on to your instance's guest OS.
3. Register the volume with the `iscsiadm` tool.

   ```
   iscsiadm -m node -o new -T <volume IQN> -p <iSCSI IP address>:<iSCSI port>
   ```

   A successful registration response resembles the following:

   ```
   New iSCSI node [tcp:[hw=,ip=,net_if=,iscsi_if=default] 169.254.0.2,3260,-1 iqn.2015-
   12.us.oracle.com:c6acda73-90b4-4bbb-9a75-faux09015418] added
   ```

4. Configure iSCSI to automatically connect to the authenticated block storage volumes after a reboot:

   ```
   iscsiadm -m node -T <volume IQN> -o update -n node.startup -v automatic
   ```

   Note: All command arguments are essential. Success returns no response.

5. Skip this step if CHAP is not enabled. If you enabled CHAP when you attached the volume, authenticate the iSCSI connection by providing the volume's CHAP credentials as follows:

   ```
   iscsiadm -m node -T <volume IQN> -p <iSCSI IP address>:<iSCSI port> -o update -n
   node.session.auth.authmethod -v CHAP
   ```

   ```
   iscsiadm -m node -T <volume IQN> -p <iSCSI IP address>:<iSCSI port> -o update -n
   node.session.auth.username -v <CHAP user name>
   ```

   ```
   iscsiadm -m node -T <volume's IQN> -p <iSCSI IP address>:<iSCSI port> -o update -n
   node.session.auth.password -v <CHAP password>
   ```

   Success returns no response.

6. Log in to iSCSI:

```
iscsiadm -m node -T <volume's IQN> -p <iSCSI IP Address>:<iSCSI port> -l
```

A successful login response resembles the following:

```
Logging in to [iface: default, target: iqn.2015-12.us.oracle.com:c6acda73-90b4-4bbb-9a75-
faux09015418, portal: 169.254.0.2,3260] (multiple)
Login to [iface: default, target: iqn.2015-12.us.oracle.com:c6acda73-90b4-4bbb-9a75-faux09015418,
portal: 169.254.0.2,3260] successful.
```

7. You can now format (if needed) and mount the volume. To get a list of mountable iSCSI devices on the instance, run the following command:

```
fdisk -l
```

The connected volume listing resembles the following:

```
Disk /dev/sdb: 274.9 GB, 274877906944 bytes, 536870912 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
```



Tip

If you have multiple volumes that do not have CHAP enabled, you can log in to them all at once by using the following commands:

```
iscsiadm -m discovery -t sendtargets -p <iSCSI IP
address>:<iSCSI port>
iscsiadm -m node -l
```

Connecting to a Volume on a Windows Instance



Warning

> When connecting to a Windows boot volume as a data volume from a second instance, you need to append `-IsMultipathEnabled $True` to the `Connect-IscsiTarget` command. See Attaching a Windows boot volume as a data volume to another instance fails for more information.

1. Use the Console to obtain the iSCSI data you need to connect the volume:
    a. Log on to Oracle Cloud Infrastructure.
    b. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
    c. Click your instance's name to display the instance details.
    d. In the **Resources** section on the **Instance Details** page, click **Attached Block Volumes** to view the attached block volume.
    e. Click the Actions icon (three dots) next to the volume you're interested in, and then click **iSCSI Commands and Information**.
    The **iSCSI Commands and Information** dialog box displays your volume's IP address and port, which you'll need to know later in this procedure.
2. Log in to your instance using a Remote Desktop client.
3. On your Windows instance, open the iSCSI Initiator. The steps to open the iSCSI Initiator may vary depending on the version of Windows.
    For example: Open **Server Manager**, click **Tools**, and then select **iSCSI Initiator**.
4. In the iSCSI Initiator Properties dialog box, click the **Discovery** tab, and then click **Discover Portal**.
5. Enter the block volume **IP Address** and **Port**, and then click **OK**.
6. Click the **Targets** tab.
7. Under **Discovered targets**, select the volume IQN.
8. Click **Connect**.

9. Make sure that the **Add this connection to the list of favorite targets** check box is selected, and then click **OK**.
10. You can now format (if needed) and mount the volume. To view a list of mountable iSCSI devices on your instance, in **Server Manager**, click **File and Storage Services**, and then click **Disks**.
    The disk is displayed in the list.

fstab Options for Block Volumes Using Consistent Device Paths

On Linux instances, if you want to automatically mount volumes on instance boot, you need to set some specific options in the `/etc/fstab` file, or the instance may fail to launch.



Note

These steps are for block volumes that are attached with consistent device paths enabled. If the block volume does not have consistent device paths enabled, use the legacy etc/fstab options instead.

Prerequisites

Before using a consistent device path, you should confirm that the instance supports consistent device paths and is correctly configured.

To verify that the volume is attached to a supported instance, connect to the instance and run the following command:

```
ll /dev/oracleoci/oraclevd*
```

The output will look similar to the following:

```
lrwxrwxrwx. 1 root root 6 Feb  7 21:02 /dev/oracleoci/oraclevda -> ../sda
lrwxrwxrwx. 1 root root 7 Feb  7 21:02 /dev/oracleoci/oraclevda1 -> ../sda1
lrwxrwxrwx. 1 root root 7 Feb  7 21:02 /dev/oracleoci/oraclevda2 -> ../sda2
lrwxrwxrwx. 1 root root 7 Feb  7 21:02 /dev/oracleoci/oraclevda3 -> ../sda3
```

If you don't see this output and instead see the following error message:

```
cannot access /dev/oracleoci/oraclevd*: No such file or directory
```

there may be a problem with the instance configuration for device paths. For assistance with this, contact Support.

Use the _netdev and nofail Options

By default, the `/etc/fstab` file is processed before the initiator starts. To configure the mount process to initiate before the volumes are mounted, specify the `_netdev` option on each line of the `/etc/fstab` file.

When you create a custom image of an instance where the volumes, excluding the root volume, are listed in the `/etc/fstab` file, instances will fail to launch from the custom image. To prevent this issue, specify the `nofail` option in the `/etc/fstab` file.

In the example scenario with three volumes, the `/etc/fstab` file entries for the volumes with the `_netdev` and `nofail` options are as follows:

```
/dev/oracleoci/oraclevdb /mnt/vol1 xfs defaults,_netdev,nofail 0 2
/dev/oracleoci/oraclevdc /mnt/vol2 xfs defaults,_netdev,nofail 0 2
/dev/oracleoci/oraclevdd /mnt/vol3 xfs defaults,_netdev,nofail 0 2
```

After you have updated the `/etc/fstab` file, use the following command to mount the volumes:

```
bash-4.2$ sudo mount -a
```

Reboot the instance to confirm that the volumes are mounted properly on reboot with the following command:

```
bash-4.2$ sudo reboot
```

Troubleshooting Issues with the /etc/fstab File

If the instance fails to reboot after you update the `/etc/fstab` file, you may need to undo the changes to the `/etc/fstab` file. To update the file, first connect to the serial console for the instance. When you have access to the instance using the serial console connection, you can remove, comment out, or fix the changes that you made to the `/etc/fstab` file.

Traditional fstab Options

On Linux instances, if you want to automatically mount volumes on instance boot, you need to set some specific options in the `/etc/fstab` file, or the instance may fail to launch.



Note

These steps are for block volumes that do not have consistent device paths enabled. If consistent device paths are enabled for the block volume, use the /etc/fstab options for block volumes using consistent device paths instead.

Volume UUIDs

On Linux operating systems, the order in which volumes are attached is non-deterministic, so it can change with each reboot. If you refer to a volume using the device name, such as `/dev/sdb`, and you have more than one non-root volume, you can't guarantee that the volume you intend to mount for a specific device name will be the volume mounted.

To prevent this issue, specify the volume UUID in the `/etc/fstab` file instead of the device name. When you use the UUID, the mount process matches the UUID in the superblock with the mount point specified in the `/etc/fstab` file. This process guarantees that the same volume is always mounted to the same mount point.

Determining the UUID for a Volume

1. Follow the steps to attach a volume and connect to the volume.
2. After the volumes are connected, create the file system of your choice on each volume using standard Linux tools.

   The remaining steps assume that three volumes were connected, and that an XFS file system was created on each volume.

3. Run the following command to use the **blkid** utility to get the UUIDs for the volumes:

```
sudo blkid
```

The output will look similar to the following:

```
{{ /dev/sda3: UUID="1701c7e0-7527-4338-ae9f-672fd8d24ec7" TYPE="xfs" PARTUUID="82d2ba4e-4d6e-
4a33-9c4d-ba52db57ea61"}}
{{ /dev/sda1: UUID="5750-10A1" TYPE="vfat" PARTLABEL="EFI System Partition" PARTUUID="082c26fd-
85f5-4db2-9f4e-9288a3f3e784"}}
{{ /dev/sda2: UUID="1aad7aca-689d-4f4f-aff0-e0d46fc1b89f" TYPE="swap" PARTUUID="94ee5675-a805-
49b2-aaf5-2fa15aade8d5"}}
{{ /dev/sdb: UUID="699a776a-3d8d-4c88-8f46-209101f318b6" TYPE="xfs"}}
{{ /dev/sdd: UUID="85566369-7148-4ffc-bf97-50954cae7854" TYPE="xfs"}}
{{ /dev/sdc: UUID="ba0ac1d3-58cf-4ff0-bd28-f2df532f7de9" TYPE="xfs"}}
```

The root volume in this output is `/dev/sda*`. The additional remote volumes are:

- `/dev/sdb`
- `/dev/sdc`
- `/dev/sdd`

4. To automatically attach the volumes at `/mnt/vol1`, `/mnt/vol2`, and `/mnt/vol3` respectively, create the three directories using the following commands:

```
bash-4.2$ sudo mkdir /mnt/vol1
{{ bash-4.2$ sudo mkdir /mnt/vol2}}
{{ bash-4.2$ sudo mkdir /mnt/vol3}}
```

Use the _netdev and nofail Options

By default, the `/etc/fstab` file is processed before the initiator starts. To configure the mount process to initiate before the volumes are mounted, specify the `_netdev` option on each line of the `/etc/fstab` file.

When you create a custom image of an instance where the volumes, excluding the root volume, are listed in the `/etc/fstab` file, instances will fail to launch from the custom image. To prevent this issue, specify the `nofail` option in the `/etc/fstab` file.

In the example scenario with three volumes, the `/etc/fstab` file entries for the volumes with the `_netdev` and `nofail` options are as follows:

```
UUID=699a776a-3d8d-4c88-8f46-209101f318b6 /mnt/vol1 xfs defaults,_netdev,nofail 0 2
UUID=ba0ac1d3-58cf-4ff0-bd28-f2df532f7de9 /mnt/vol2 xfs defaults,_netdev,nofail 0 2
UUID=85566369-7148-4ffc-bf97-50954cae7854 /mnt/vol3 xfs defaults,_netdev,nofail 0 2
```

After you have updated the `/etc/fstab` file, use the following command to mount the volumes:

```
bash-4.2$ sudo mount -a
```

Reboot the instance to confirm that the volumes are mounted properly on reboot with the following command:

```
bash-4.2$ sudo reboot
```

Troubleshooting Issues with the /etc/fstab File

If the instance fails to reboot after you update the `/etc/fstab` file, you may need to undo the changes to the `/etc/fstab` file. To update the file, first connect to the serial console for the instance. When you have access to the instance using the serial console connection, you can remove, comment out, or fix the changes that you made to the `/etc/fstab` file.

Listing Volumes

You can list all Block Volume volumes in a specific compartment, as well as detailed information on a single volume.

Required IAM Service Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let users launch Compute instances includes the ability to list volumes. The policy in Let volume admins manage block volumes, backups, and volume groups lets the specified group do everything with block volumes and backups, but not launch instances.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Using the Console

Open the navigation menu. Under **Core Infrastructure**, go to **Block Storage** and click **Block Volumes**. A detailed list of volumes in your current compartment is displayed.

- To view the volumes in a different compartment, change the compartment in the **Compartment** drop-down menu.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

List Volumes:

Get a list of volumes within a compartment.

- ListVolumes

Get a Single Volume:

Get detailed information on a single volume:

- GetVolume

Listing Volume Attachments

You can use the API to list all Block Volume volume attachments in a specific compartment, as well as detailed information on a single volume attachment.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let users launch Compute instances includes the ability to list volume attachments. The policy in Let volume admins manage block volumes, backups, and volume groups lets the specified group do everything with block volumes and backups, but not launch instances.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

List Attachments:

Get information on all volume attachments in a specific compartment.

- ListVolumeAttachments

Get a Single Attachment:

Get detailed information on a single attachment.

- GetVolumeAttachment

Listing Boot Volume Attachments

You can use the API to list all the boot volume attachments in a specific compartment. You can also use the API to retrieve detailed information on a single boot volume attachment.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let users launch Compute instances includes the ability to list volume attachments. The policy in Let volume admins manage block volumes, backups, and volume groups lets the specified group do everything with block volumes and backups, but not launch instances.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

List Boot Volume Attachments:

Get information on all boot volume attachments in a specific compartment.

- ListBootVolumeAttachments

Get a Single Boot Volume Attachment:

Get detailed information on a single boot volume attachment.

- GetBootVolumeAttachment

Renaming a Volume

You can use the API to change the display name of a Block Volume volume.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let users launch Compute instances includes the ability to rename block volumes. The policy in Let volume admins manage block volumes, backups, and volume groups lets the specified group do everything with block volumes and backups, but not launch instances.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Using the API

To update a volume's display name, use the following operation:

- UpdateVolume

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Resizing a Volume

The Oracle Cloud Infrastructure Block Volume service lets you expand the size of block volumes and boot volumes. You have three options to increase the size of your volumes:

- Expand an existing volume in place with offline resizing. See Resizing a Volume Using the Console for the steps to do this.
- Restore from a volume backup to a larger volume. See Restoring a Backup to a New Volume and Restoring a Boot Volume.
- Clone an existing volume to a new, larger volume. See Cloning a Volume and Cloning a Boot Volume.

For more information about the Block Volume service, see the Block Volume FAQ.

You can only increase the size of the volume, you cannot decrease the size. You can attach a volume and start using it as soon as it's resized and becomes available.

This topic describes how to expand your volume in place with offline resizing.



Warning

Before you resize a boot or block volume, you should create a backup of the volume.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK,

CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let users launch Compute instances includes the ability to attach/detach existing block volumes. The policy in Let volume admins manage block volumes, backups, and volume groups lets the specified group do everything with block volumes and backups, but not launch instances.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Considerations When Resizing a Volume

Whenever you detach and re-attach volumes, there are complexities and risks for both Linux-based and Windows-based instances. This applies to both paravirtualized and iSCSI attachment types. You should keep the following in mind when resizing volumes:

- When you re-attach a volume to an instance after resizing, if you are not using consistent device paths, or the instance does not support consistent device paths, device order and path may change. If you are using a tool such as Logical Volume Manager (LVM), you may need to fix the device mappings. For more information about consistent device paths, see Connecting to Volumes With Consistent Device Paths.
- When you detach and then re-attach an iSCSI-attached volume to an instance, the volume's IP address will increment.
- Before you resize a volume, you should create a full backup of the volume.

Resizing a Volume Using the Console

Resizing a Block Volume

1. Detach the block volume, see Detaching a Volume.

2. Open the navigation menu. Under **Core Infrastructure**, go to **Block Storage** and click **Block Volumes**.
3. In the **Block Volumes** list, click the block volume you want to resize.
4. Click **Resize**.
5. Specify the new size and click **Resize**. You must specify a larger value than the block volume's current size.
6. Reattach the block volume, see Attaching a Volume.
7. Extend the partition, see Extending the Partition for a Block Volume.

Resizing a Boot Volume for a Windows Instance

1. Stop the instance, see Stopping and Starting an Instance.
2. Detach the boot volume, see Detaching a Boot Volume.
3. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Boot Volumes**.
4. In the **Boot Volumes** list, click the boot volume you want to resize.
5. Click **Resize**.
6. Specify the new size and click **Resize**. You must specify a larger value than the boot volume's current size.
7. Reattach the boot volume, see Attaching a Boot Volume.
8. Restart the instance, see Stopping and Starting an Instance.
9. Extend the partition, see Extending the System Partition on a Windows-Based Image.

Resizing a Boot Volume for a Linux Instance

1. Stop the instance, see Stopping and Starting an Instance.
2. Detach the boot volume, see Detaching a Boot Volume.

3. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Boot Volumes**.
4. In the **Boot Volumes** list, click the boot volume you want to resize.
5. Click **Resize**.
6. Specify the new size and click **Resize**. You must specify a larger value than the boot volume's current size.
7. Attach the boot volume to a second instance as a data volume. See Attaching a Volume and Connecting to a Volume.
8. Extend the partition and grow the file system, see Extending the Root Partition on a Linux-Based Image.
9. Reattach the boot volume, see Attaching a Boot Volume.
10. Restart the instance, see Stopping and Starting an Instance.

Extending the Partition for a Block Volume

The Oracle Cloud Infrastructure Block Volume service lets you expand the size of block volumes with offline volume resizing. For more information, see Resizing a Volume. In order to take advantage of the larger volume size, you need to extend the partition for the block volume.

Required IAM Policy

Extending a partition on an instance does not require a specific IAM policy. However, you may need permission to run the necessary commands on the instance's guest OS. Contact your system administrator for more information.

Extending a Partition on a Linux-Based Image

On Linux-based images, use the following steps to extend the partition for a block volume.

PREREQUISITES

After you have resized a volume, you need to attach it to an instance before you can extend the partition and grow the file system. See Attaching a Volume and Connecting to a Volume for more information.

EXTENDING THE LINUX PARTITION

Extending a partition

1. To identify the volume that you want to extend the partition for, run the following command to list the attached block volumes:

   ```
   lsblk
   ```

2. Run the following command to edit the volume's partition table with `parted`:

   ```
   parted <volume_id>
   ```

   <volume_id> is the volume identifier, for example `/dev/sdc`.

3. When you run `parted`, you may encounter the following error message:

   ```
   Warning: Not all of the space available to <volume_id> appears to be used,
   you can fix the GPT to use all of the space (an extra volume_size blocks)
   or continue with the current setting?
   ```

   You are then prompted to fix the error or ignore the error and continue with the current setting. Specify the option to fix the error.

4. Run the following command to change the display units to sectors so that you can see the precise start position for the volume:

   ```
   (parted) unit s
   ```

5. Run the following command to display the current partitions in the partition table:

   ```
   (parted) print
   ```

   Make note of the values in the **Number**, **Start**, and **File system** columns for the root partition.

6. Run the following command to remove the existing root partition:

   ```
   (parted) rm <partition_number>
   ```

   <partition_number> is the value from the **Number** column.

7. Run the following command to recreate the partition:

   ```
   (parted) mkpart
   ```

   At the `Start?` prompt, specify the value from the **Start** column. At the `File system type?` prompt, specify the value from the **File system** column. Specify `100%` for the `End?` prompt.

8. Run the following command to exit `parted`:

   ```
   (parted) quit
   ```

   This command forces a rewrite of the partition table with the new partition settings that you specified.

9. To verify that the root partition was extended, run the following command to list the attached block volumes:

   ```
   lsblk
   ```

After you extend the root partition you need to grow the file system. The steps in the following procedure apply only to xfs file systems.

Growing the file system for a partition

1. Before you grow the file system, repair any issues with the file system on the extended partition by running the following command:

   ```
   xfs_repair <partition_id>
   ```

   <partition_id> is the partition identifier, for example `/dev/sdc1`. See Checking and Repairing an XFS File System for more information.

2. After you have confirmed that there are no more issues to repair, you need to create a mount point to run the `xfs_growfs` against. To do this, create a directory and mount the partition to that directory by running the following commands:

   ```
   mkdir <directory_name>
   mount <partition_id> <directory_name> -o nouuid
   ```

   <partition_id> is the partition identifier, for example `/dev/sdc1`, and *<directory_name>* is the directory name, for example `data`.

3. After you have created the mount point run the following command to grow the file system:

   ```
   xfs_growfs -d <directory_name>
   ```

   <directory_name> is the name for the directory you created in the previous step, for example `data`.

4. To verify that the file system size is correct, run the following command to display the file system details:

   ```
   df -lh
   ```

Extending a Partition on a Windows-Based Image

On Windows-based images, you can extend a partition using the Windows interface or from the command line using the DISKPART utility.

Windows Server 2016 and Windows Server 2012

The steps to extend a partition for a block volume attached to an instance running Windows 2012 or Windows 2016 are the same, and are described in the following procedures.

Extending a partition using the Windows interface

1. Open the Disk Management system utility on the instance.
2. Right-click the expanded block volume and select **Extend Volume**.

3. Follow the instructions in the **Extend Volume Wizard**:
    a. Select the disk that you want to extend, enter the size, and then click **Next**.
    b. Confirm that the disk and size settings are correct, and then click **Finish**.
4. Verify that the block volume's disk has been extended in Disk Management.

Extending a partition using the command line with DISKPART

1. Open a command prompt as administrator on the instance.
2. Run the following command to start the DISKPART utility:

```
diskpart
```

3. At the `DISKPART` prompt, run the following command to display the instance's volumes:

```
list volume
```

4. Run the following command to select the expanded block volume:

```
select volume <volume_number>
```

<volume_number> is the number associated with the block volume that you want to extend the partition for.

5. Run the following command to extend the partition:

```
extend size=<increased_size_in_MB>
```

<increased_size_in_MB> is the size in MB that you want to extend the partition to.



Warning

When using the DISKPART utility, do not overextend the partition beyond the current available space. Overextending the partition could result in data loss.

6. To confirm that the partition was extended, run the following command and verify that the block volume's partition has been extended:

```
list volume
```

WINDOWS SERVER 2008

Use the steps described in the following procedures to extend a partition on instances running Windows 2008.

Extending the system partition using the Windows interface

1. Open the Server Manager on the instance.
2. Expand the **Storage** node in the left navigation pane and click **Disk Management**.
3. Right-click the expanded block volume and select **Extend Volume**.
4. Follow the instructions in the **Extend Volume Wizard**:
    a. Select the disk that you want to extend, enter the size, and then click **Next**.
    b. Confirm that the disk and size settings are correct, and then click **Finish**.
5. Verify that the block volume's disk has been extended in the Server Manager's **Disk Management** node.

Extending a partition using the command line with DISKPART

1. Open a command prompt as administrator on the instance.
2. Run the following command to start the DISKPART utility:

```
diskpart
```

3. At the `DISKPART` prompt, run the following command to display the instance's volumes:

```
list volume
```

4. Run the following command to select the expanded block volume:

```
select volume <volume_number>
```

   <volume_number> is the number associated with the block volume that you want to extend the partition for.

5. Run the following command to extend the partition:

```
extend size=<increased_size_in_MB>
```

   <increased_size_in_MB> is the size in MB that you want to extend the partition to.



Warning

When using the DISKPART utility, do not overextend the partition beyond the current available space. Overextending the partition could result in data loss.

6. To confirm that the partition was extended, run the following command and verify that the boot volume's partition has been extended:

```
list volume
```

Overview of Block Volume Backups

The backups feature of the Oracle Cloud Infrastructure Block Volume service lets you make a point-in-time backup of data on a block volume. These backups can then be restored to new volumes either immediately after a backup or at a later time that you choose.

Backups are encrypted and stored in Oracle Cloud Infrastructure Object Storage, and can be restored as new volumes to any availability domain within the same region they are stored. This capability provides you with a spare copy of a volume and gives you the ability to successfully complete disaster recovery within the same region.

There are two ways you can initiate a backup, either by manually starting the backup, or by assigning a policy which defines a set backup schedule.

Manual Backups

These are on-demand one-off backups that you can launch immediately by following the steps described in Backing Up a Volume. When launching a manual backup, you can specify whether an incremental or a full backup should be performed. See Volume Backup Types for more information about backup types.

Policy-Based Backups

These are automated scheduled backups. Each backup policy has a set backup frequency and retention period. There are three predefined policies, Bronze, Silver, and Gold.

See Policy-Based Backups for more information.

Volume Backup Types

There are two backup types available in the Block Volume service:

- **Incremental:** This backup type includes only the changes since the last backup.
- **Full:** This backup type includes all changes since the volume was created.



Note

Backup Details

Backups are not an identical copy of the volume being backed up. For incremental backups, they are a record of all the changes since the last backup. For full backups, they are a record of all the changes since the volume was created. For example, in a scenario where you create a 16 TB block volume, modify 40 GB on the volume, and then launch a full backup, upon completion the volume backup size is 40 GB.

Planning Your Backup

The primary use of backups is to support business continuity, disaster recovery, and long-term archiving requirements. When determining a backup schedule, your backup plan and goals should consider the following:

- **Frequency:** How often you want to back up your data.
- **Recovery time:** How long you can wait for a backup to be restored and accessible to the applications that use it. The time for a backup to complete varies on several factors, but it will generally take a few minutes or longer, depending on the size of the data being backed up and the amount of data that has changed since your last backup.
- **Number of stored backups:** How many backups you need to keep available and the deletion schedule for those you no longer need. You can only create one backup at a time, so if a backup is underway, it will need to complete before you can create another one. For details about the number of backups you can store, see Block Volume Capabilities and Limits.

The common use cases for using backups are:

- Needing to create multiple copies of the same volume. Backups are highly useful in cases where you need to create many instances with many volumes that need to have the same data formation.
- Taking a snapshot of your work that you can restore to a new volume at a later time.
- Ensuring you have a spare copy of your volume in case something goes wrong with your primary copy.

Copying Block Volume Backups Across Regions

You can copy block volume backups between regions using the Console, command line interface (CLI), SDKs, or REST APIs. For steps, see Copying a Volume Backup Between Regions. This capability enhances the following scenarios:

- **Disaster recovery and business continuity:** By copying block volume backups to another region at regular intervals, it makes it easier for you to rebuild applications and data in the destination region if a region-wide disaster occurs in the source region.
- **Migration and expansion:** You can easily migrate and expand your applications to another region.

To copy volume backups between regions, you must have permission to read and copy volume backups in the source region, and permission to create volume backups in the destination region. For more information see Required IAM Policy.

Once you have copied the volume backup to the new region you can then restore from that backup by creating a new volume from the backup using the steps described in Restoring a Backup to a New Volume.

Volume Backup Encryption

The Oracle Cloud Infrastructure Block Volume service always encrypts all block volumes, boot volumes, and volume backups at rest by using the Advanced Encryption Standard (AES) algorithm with 256-bit encryption.

The Oracle Cloud Infrastructure Key Management service enables you to bring and manage your own keys to use for encrypting volumes and their backups. When you create a volume backup, the encryption key used for the volume is also used for the volume backup. When you restore the backup to create a new volume you configure a new key, see Restoring a Backup to a New Volume. See also Overview of Key Management.

If you do not configure a volume to use the Key Management service, the Block Volume service uses the Oracle-provided encryption key instead. This applies to both encryption at-rest and in-transit encryption.

Best Practices When Creating Block Volume Backups

When creating and restoring from backups, keep in mind the following:

- Before creating a backup, you should ensure that the data is consistent: Sync the file system, unmount the file system if possible, and save your application data. Only the data on the disk will be backed up. When creating a backup, after the backup state changes from REQUEST_RECEIVED to CREATING, you can return to writing data to the volume. While a backup is in progress, the volume that is being backed up cannot be deleted.
- If you want to attach a restored volume that has the original volume attached, be aware that some operating systems do not allow you to restore identical volumes. To resolve this, you should change the partition IDs before restoring the volume. The steps to change an operating system's partition ID vary by operating system. For instructions, see your operating system's documentation.
- You should not delete the original volume until you have verified that the backup you created of it completed successfully.

See Backing Up a Volume and Restoring a Backup to a New Volume for more information.

Differences Between Block Volume Backups and Clones

Consider the following criteria when you decide whether to create a backup or a clone of a volume.

	Volume Backup	**Volume Clone**
Description	Creates a point-in-time backup of data on a volume. You can restore multiple new volumes from the backup later in the future.	Creates a single point-in-time copy of a volume without having to go through the backup and restore process.
Use case	Retain a backup of the data in a volume, so that you can duplicate an environment later or preserve the data for future use. Meet compliance and regulatory requirements, because the data in a backup remains unchanged over time. Support business continuity requirements. Reduce the risk of outages or data mutation over time.	Rapidly duplicate an existing environment. For example, you can use a clone to test configuration changes without impacting your production environment.
Speed	Slower (minutes or hours)	Faster (seconds)
Cost	Lower cost	Higher cost
Storage location	Object Storage	Block Volume

	Volume Backup	Volume Clone
Retention policy	Policy-based backups expire, manual backups do not expire	No expiration
Volume groups	Supported. You can back up a volume group.	Supported. You can clone a volume group.

For background information and steps to clone a block volume, see Cloning a Volume.

Using the CLI or REST APIs to Customize and Manage the Lifecycle of Volume Backups

You can use the CLI, REST APIs, or the SDKs to automate, script, and manage volume backups and their lifecycle.

Using the CLI

This section provides basic sample CLI commands that you can use in a script, such as a cron job run by the cron utility on Linux-based operating systems, to perform automatic backups at specific times. For information about using the CLI, see Command Line Interface (CLI).

To create a manual backup of the specified block volume

Open a command prompt and run:

```
oci bv backup create --volume-id <block_volume_OCID> --display-name <Name> --type <FULL|INCREMENTAL>
```

For example:

```
oci bv backup create --volume-id  ocid1.volume.oc1..<unique_ID> --display-name "backup display name" --
type FULL
```

To delete a block volume backup

Open a command prompt and run:

```
oci bv backup delete --volume-backup-id <volume_backup_OCID>
```

For example:

```
oci bv backup delete --volume-backup-id ocid1.volume.oc1..<unique_ID>
```

To create a manual backup of the specified boot volume

Open a command prompt and run:

```
oci bv boot-volume-backup create --volume-id <boot_volume_OCID> --display-name <Name> --type
<FULL|INCREMENTAL>
```

For example:

```
oci bv boot-volume-backup create --volume-id  ocid1.volume.oc1..<unique_ID> --display-name "backup
display name" --type FULL
```

To delete a boot volume backup

Open a command prompt and run:

```
oci bv backup delete --boot-volume-backup-id <boot_volume__backup_OCID>
```

For example:

```
oci bv backup delete --boot-volume-backup-id ocid1.volume.oc1..<unique_ID>
```

To list the Oracle-defined backup policies

Open a command prompt and run:

```
oci bv volume-backup-policy list
```

To assign an Oracle-defined backup policy to a boot or block volume

Open a command prompt and run:

```
oci bv volume-backup-policy-assignment create --asset-id <volume_OCID> --policy-id <policy_OCID>
```

For example:

```
oci bv volume-backup-policy-assignment create --asset-id  ocid1.volume.oc1..<unique_ID> --policy-id
ocid1.volumebackuppolicy.oc1..<unique_ID>
```

To un-assign an Oracle-defined backup policy from a boot or block volume

Open a command prompt and run:

```
oci bv volume-backup-policy-assignment delete --policy-assignment-id <policy_assignment_OCID>
```

For example:

```
oci bv volume-backup-policy-assignment delete --policy-assignment-id
ocid1.volumebackuppolicyassign.oc1..<unique_ID>
```

To retrieve the backup policy assignment ID for a boot or block volume

Open a command prompt and run:

```
oci bv volume-backup-policy-assignment get-volume-backup-policy-asset-assignment --asset-id <volume_
OCID>
```

For example:

```
oci bv volume-backup-policy-assignment get-volume-backup-policy-asset-assignment --asset-id
ocid1.volume.oc1..<unique_ID>
```

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operations for working with block volume backups, boot volume backups, and backup policies.

Block Volume Backups

- CreateVolumeBackup
- DeleteVolumeBackup
- GetVolumeBackup
- ListVolumeBackups
- UpdateVolumeBackup

Boot Volume Backups

- CreateBootVolumeBackup
- DeleteBootVolumeBackup
- GetBootVolumeBackup
- ListBootVolumeBackups
- UpdateBootVolumeBackup

Volume Backup Policies and Policy Assignments

- GetVolumeBackupPolicy
- ListVolumeBackupPolicies
- CreateVolumeBackupPolicyAssignment
- DeleteVolumeBackupPolicyAssignment
- GetVolumeBackupPolicyAssetAssignment
- GetVolumeBackupPolicyAssignment

Backing Up a Volume

You can create a backup of a volume using Block Volume.

For information to help you decide whether to create a backup or a clone of a boot volume, see Differences Between Block Volume Backups and Clones.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let volume admins manage block volumes, backups, and volume groups lets the specified group do everything with block volumes and backups. The policy in Let volume backup admins manage only backups further restricts access to just creating and managing backups.



Tip

When users create a backup from a volume or restore a volume from a backup, the volume and backup don't have to be in the same compartment. However, users must have access to both compartments.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Block Storage** and click **Block Volumes**.
2. Click the block volume that you want to create a backup for.
3. Click **Create Manual Backup**.
4. Enter a name for the backup.
5. Select the backup type, either incremental or full. See Volume Backup Types for

information about backup types.

6. Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
7. Click **Create Backup**.
   The backup will be completed once its icon no longer lists it as **CREATING** in the volume list.

Using the API

To back up a volume, use the following operation:

- CreateVolumeBackup

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

For more information about backups, see Overview of Block Volume Backups and Restoring a Backup to a New Volume.

Policy-Based Backups

The Oracle Cloud Infrastructure Block Volume service provides you with the capability to perform volume backups automatically on a schedule and retain them based on the selected backup policy. This allows you to adhere to your data compliance and regulatory requirements.



Warning

Deleting Block Volumes with Policy-Based Backups

All policy-based backups will eventually expire, so if you want to keep a volume backup indefinitely, you need to create a manual backup.

Volume Backup Policies

There are three predefined backup policies, Bronze, Silver, and Gold. Each backup policy has a set backup frequency and retention period.

Bronze Policy

The bronze policy includes monthly incremental backups, run on the first day of the month. These backups are retained for twelve months. This policy also includes a full backup, run yearly on January 1st. Full backups are retained for five years.

Silver Policy

The silver policy includes weekly incremental backups that run on Sunday. These backups are retained for four weeks. This policy also includes monthly incremental backups, run on the first day of the month and are retained for twelve months. Also includes a full backup, run yearly on January 1st. Full backups are retained for five years.

Gold Policy

The gold policy includes daily incremental backups. These backups are retained for seven days. This policy also includes weekly incremental backups that run on Sunday and are retained for four weeks. Also includes monthly incremental backups, run on the first day of the month, retained for twelve months, and a full backup, run yearly on January 1st. Full backups are retained for five years.

Assigning a Backup Policy to Volumes

When you create a new volume on Oracle Cloud Infrastructure you can select the appropriate backup policy at that time, for more information, see Creating a Volume.

If your requirements change, you can adjust the schedule and retention period by selecting a different backup policy or by removing it all together. You can also assign a backup policy to an existing volume. See Using the Console.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.



Important

To view or work with backup policies, you need access to the root compartment, which is where the predefined backup policies are located.

For administrators: The policy in Let volume admins manage block volumes, backups, and volume groups lets the specified group do everything with block volumes and backups. The policy in Let volume backup admins manage only backups further restricts access to just creating and managing backups.



Tip

When users create a backup from a volume or restore a volume from a backup, the volume and backup don't have to be in the same compartment. However, users must have access to both compartments.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Using the Console

You can use the Console to assign or change the backup policy for an existing volume.

To assign a backup policy

1. Open the navigation menu. Under **Core Infrastructure**, go to **Block Storage** and click **Block Volumes**.
2. Click the volume for which you want to assign a backup policy to.
3. Click **Assign Backup Policy**.
4. Select the appropriate backup policy for your requirements.
5. Click **Assign Backup Policy**.

To remove a backup policy

1. Open the navigation menu. Under **Core Infrastructure**, go to **Block Storage** and click **Block Volumes**.
2. Click the volume for which you want to remove the backup policy for.
3. Click **Remove Backup Policy**.

4. Click **OK** to confirm the backup policy removal.

To change a backup policy

1. Open the navigation menu. Under **Core Infrastructure**, go to **Block Storage** and click **Block Volumes**.
2. Click the volume for which you want to change the backup policy for.
3. Click **Remove Backup Policy**.
4. Click **OK** to confirm the backup policy removal.
5. Click **Assign Backup Policy**.
6. Select the backup policy you want to switch to.
7. Click **Assign Backup Policy**.

For more information about backups, see Overview of Block Volume Backups and Restoring a Backup to a New Volume.

Copying a Volume Backup Between Regions

You can copy volume backups from one region to another region using the Oracle Cloud Infrastructure Block Volume service. For more information, see Copying Block Volume Backups Across Regions.



Note

Limitations for Copying Volume Backups Across Regions

Copying backups for block volumes encrypted using Key Management across regions is not supported.

Copying boot volume backups across regions is not supported.

When copying block volume backups across regions in your tenancy, you can only copy one backup at a time from a specific source region.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The first two statements listed in the Let volume admins manage block volumes, backups, and volume groups policy lets the specified group do everything with block volumes and backups with the exception of copying volume backups across regions. The aggregate resource type `volume-family` does not include the `VOLUME_BACKUP_COPY` permission, so to enable copying volume backups across regions you need to ensure that you include the third statement in that policy, which is:

```
Allow group VolumeAdmins to copy volume-backups in tenancy
```

To restrict access to just creating and managing volume backups, including copying volume backups between regions, use the policy in Let volume backup admins manage only backups. The individual resource type `volume-backups` includes the `VOLUME_BACKUP_COPY` permission, so you do not need to specify it explicitly in this policy.

Restricting Access

The specific permissions needed to copy volume backups across regions are:

- **Source region**: `VOLUME_BACKUP_READ`, `VOLUME_BACKUP_COPY`
- **Destination region**: `VOLUME_BACKUP_CREATE`

Sample Policies

To restrict a group to specific source and destination regions for copying volume backups

In this example, the group is restricted to copying volume backups from the uk-london-1 region to the eu-frankfurt-1 region.

```
Allow group MyTestGroup to read volume-backups in tenancy where all {request.region='lhr'}
Allow group MyTestGroup to use volume-backups in tenancy where all {request.permission='VOLUME_BACKUP_
COPY', request.region = 'lhr',
Allow group MyTestGroup to manage volume-backups in tenancy where all {request.permission='VOLUME_
BACKUP_CREATE', request.region = 'fra'}
```

To restrict some source regions to specific destination regions while enabling all destination regions for other source regions

In this example, the following is enabled for the group:

- Manage volume backups in all regions.
- Copy volume backups from the us-phoenix-1 and us-ashburn-1 regions to any destination regions.
- Copy volume backups from the eu-frankfurt-1 and uk-london-1 regions only to the eu-frankfurt-1 or uk-london-1 regions.

```
Allow group MyTestGroup to read volume-backups in tenancy where all {request.region='lhr'}
Allow group MyTestGroup to manage volume-backups in tenancy where any {request.permission!='VOLUME_
BACKUP_COPY'}
Allow group MyTestGroup to use volume-backups in tenancy where all {request.permission='VOLUME_BACKUP_
```

```
COPY', any {request.region='lhr', request.region='fra'}, any{target.region='fra', target.region='lhr'}}
Allow group MyTestGroup to use volume-backups in tenancy where all {request.permission='VOLUME_BACKUP_
COPY', any {request.region='phx', request.region='iad'}}
```

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Using the Console

1. Open the navigation menu. Under **Block Storage**, click **Block Volume Backups**. A list of the block volume backups in the compartment you're viewing is displayed. If you don’t see the one you're looking for, make sure you’re viewing the correct compartment (select from the list on the left side of the page).
2. Click the Actions icon (three dots) for the block volume backup you want to copy to another region.
3. Click **Copy to Another Region**.
4. Enter a name for the backup and choose the region to copy the backup to.
5. Click **Copy Block Volume Backup**.
6. Confirm that the source and destination region details are correct in the confirmation dialog and then click **OK**.

Using the API

To copy a volume backup to another region, use the following operation:

- CopyVolumeBackup

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Next Steps

After copying the block volume backup, switch to the destination region in the Console and verify that the copied backup appears in the list of block volume backups for that region. You can then restore the backup by creating a new block volume from it using the steps in Restoring a Backup to a New Volume.

For more information about backups, see Overview of Block Volume Backups.

Restoring a Backup to a New Volume

You can restore a backup of a volume as a new volume using Block Volume.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let volume admins manage block volumes, backups, and volume groups lets the specified group do everything with block volumes and backups.



Tip

When users create a backup from a volume or restore a volume from a backup, the volume and backup don't have to be in the same compartment. However, users must have access to both compartments.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Using the Console

1. Open the navigation menu. Under **Block Storage**, click **Block Volume Backups**. A list of the block volume backups in the compartment you're viewing is displayed. If you don't see the one you're looking for, make sure you're viewing the correct compartment (select from the list on the left side of the page).
2. Click the Actions icon (three dots) for the block volume backup you want to restore.
3. Click **Create Block Volume**.
4. Enter a name for the block volume and choose the availability domain in which you want to restore it.
5. You can restore a block volume backup to a larger volume size. To do this, check **Custom Block Volume Size (GB)**, and then specify the new size. You can only increase the size of the volume, you cannot decrease the size. If you restore the block volume backup to a larger size volume, you need to extend the volume's partition, see Extending the Partition for a Block Volume for more information.
6. Optionally, you can select the appropriate backup policy for your requirements. See Policy-Based Backups for more information about backup policies.
7. Optionally, you can encrypt the data in this volume using your own Key Management encryption key. To use Key Management for your encryption needs, select the **Encrypt using Key Management** check box. Then, select the **Vault Compartment** and **Vault** that contain the master encryption key you want to use. Also select the **Master Encryption Key Compartment** and **Master Encryption Key**. For more information about encryption, see Overview of Key Management.
8. Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
9. Click **Create**.
   The volume will be ready to attach once its icon no longer lists it as **PROVISIONING** in the volume list. For more information, see Attaching a Volume.



Warning

If you want to attach a restored volume that has the original volume attached, be aware that some operating systems do not allow you to restore identical volumes. To resolve this, you should change the partition IDs before restoring the volume. How to change an operating system's partition ID varies by operating system; for instructions, see your operating system's documentation.

Using the API

The API used to restore a backup is CreateVolume. The API has an optional volumeBackupId parameter that you can use to define the backup from which the data should be restored on the newly created volume. For details, see CreateVolumeDetails Reference.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

For more information about backups, see Overview of Block Volume Backups and Backing Up a Volume.

Cloning a Volume

You can create a clone from a volume using the Block Volume service. Cloning enables you to make a copy of an existing block volume without needing to go through the backup and restore process.

A cloned volume is a point-in-time direct disk-to-disk deep copy of the source volume, so all the data that is in the source volume when the clone is created is copied to the clone volume. Any subsequent changes to the data on the source volume are not copied to the clone. Since

the clone is a copy of the source volume it will be the same size as the source volume unless you specify a larger volume size when you create the clone.

The clone operation occurs immediately, and you can attach and use the cloned volume as a regular volume as soon as the state changes to available. At this point, the volume data is being copied in the background, and can take up to thirty minutes depending on the size of the volume.

There is a single point-in-time reference for a source volume while it is being cloned, so if the source volume is attached when a clone is created, you need to wait for the first clone operation to complete from the source volume before creating additional clones. If the source volume is detached, you can create up to ten clones from the same source volume simultaneously.

You can only create a clone for a volume within the same region, availability domain and tenant. You can create a clone for a volume between compartments as long as you have the required access permissions for the operation.

For more information about the Block Volume service and cloned volumes, see the Block Volume FAQ.

Differences Between Block Volume Clones and Backups

Consider the following criteria when you decide whether to create a backup or a clone of a volume.

	Volume Backup	**Volume Clone**
Description	Creates a point-in-time backup of data on a volume. You can restore multiple new volumes from the backup later in the future.	Creates a single point-in-time copy of a volume without having to go through the backup and restore process.
Use case	Retain a backup of the data in a volume, so that you can duplicate an environment later or preserve the data for future use. Meet compliance and regulatory requirements, because the data in a backup remains unchanged over time. Support business continuity requirements. Reduce the risk of outages or data mutation over time.	Rapidly duplicate an existing environment. For example, you can use a clone to test configuration changes without impacting your production environment.
Speed	Slower (minutes or hours)	Faster (seconds)
Cost	Lower cost	Higher cost
Storage location	Object Storage	Block Volume

	Volume Backup	Volume Clone
Retention policy	Policy-based backups expire, manual backups do not expire	No expiration
Volume groups	Supported. You can back up a volume group.	Supported. You can clone a volume group.

For more information about block volume backups, see Overview of Block Volume Backups and Backing Up a Volume.

Using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Block Storage** and click **Block Volumes**.
2. In the **Block Volumes** list, click the volume that you want to clone.
3. In **Resources**, click **Clones**.
4. Click **Create Clone**.
5. Specify a name for the clone.
6. If you want to clone the block volume to a larger size volume, select **Custom Block Volume Size (GB)** and then specify the new size. You can only increase the size of the volume, you cannot decrease the size. If you clone the block volume to a larger size volume, you need to extend the volume's partition. See Extending the Partition for a Block Volume for more information.
7. Click **Create Clone**.

The volume is ready use when its icon lists it as **AVAILABLE** in the volume list. At this point, you can perform various actions on the volume such as creating a clone from the volume, attaching it to an instance, or deleting the volume.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

To create a clone from a volume, use the CreateVolume operation and specify VolumeSourceFromVolumeDetails for CreateVolumeDetails.

Disconnecting From a Volume

For volumes attached with iSCSI as the volume attachment type you need to disconnect the volume from an instance before you detach the volume. For more information about attachment type options, see Volume Attachment Types.

Required IAM Policy

Disconnecting a volume from an instance does not require a specific IAM policy. Don't confuse this with detaching a volume (see Detaching a Volume).

Disconnecting from a Volume on a Linux Instance



Warning

Oracle recommends that you unmount and disconnect the volume from the instance using `iscsiadm` before you detach the volume. Failure to do so may lead to loss of data.

1. Log on to your instance's guest OS and unmount the volume.
2. Run the following command to disconnect the instance from the volume:

```
iscsiadm -m node -T <IQN> -p <iSCSI IP ADDRESS>:<iSCSI PORT> -u
```

A successful logout response resembles the following:

```
Logging out of session [sid: 2, target: iqn.2015-12.us.oracle.com:c6acda73-90b4-4bbb-9a75-
faux09015418, portal: 169.254.0.2,3260]
Logout of [sid: 2, target: iqn.2015-12.us.oracle.com:c6acda73-90b4-4bbb-9a75-faux09015418,
portal: 169.254.0.2,3260] successful.
```

3. You can now detach the volume without the risk of losing data.

Disconnecting from a Volume on a Windows Instance

1. Use a Remote Desktop client to log on to your Windows instance, and then open **Disk Management**.
2. Right-click the volume you want to disconnect, and then click **Offline**.
3. Open **iSCSI Initiator**, select the target, and then click **Disconnect**.
4. Confirm the session termination. The status should show as **Inactive**.
5. In **iSCSI Initiator**, click the **Favorite Targets** tab, select the target you are disconnecting, and then click **Remove**.
6. Click the **Volumes and Devices** tab, select the volume from the **Volume List**, and then click **Remove**.
7. You can now detach the volume without the risk of losing data.

Detaching a Volume

When an instance no longer needs access to a volume, you can detach the volume from the instance without affecting the volume's data.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have

permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let users launch Compute instances includes the ability to attach/detach existing block volumes. The policy in Let volume admins manage block volumes, backups, and volume groups lets the specified group do everything with block volumes and backups, but not launch instances.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Using the Console



Warning

For volumes attached using iSCSI Oracle recommends that you unmount and disconnect the volume from the instance using `iscsiadm` before you detach the volume. Failure to do so may lead to loss of data. See Disconnecting From a Volume for more information.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
2. In the **Instance** list locate the instance. Click its name to display the instance details.
3. In the **Resources** section on the **Instance Details** page, click **Attached Block Volumes**
4. Click **Detach** next to the volume you want to detach and confirm the selection when prompted.

Using the API

To delete an attachment, use the following operation:

- DetachVolume

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Volume

You can delete a volume that is no longer needed.



Warning

- You cannot undo this operation. Any data on a volume will be permanently deleted once the volume is deleted.
- All policy-based backups will eventually expire, so if you want to keep a volume backup indefinitely, you need to create a manual backup. See Overview of Block Volume Backups for information about policy-based and manual backups.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let volume admins manage block volumes, backups, and volume groups lets the specified group do everything with block volumes and backups.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Block Storage** and click **Block Volumes**.
2. In the **Block Volumes** list, find the volume you want to delete.
3. Click **Terminate** next to the volume you want to delete and confirm the selection when prompted.

Using the API

To delete a volume, use the following operation:

- DeleteVolume

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Block Volume Performance

The content in the sections below apply to **Category 7** and **Section 3.b** of the Oracle PaaS and IaaS Public Cloud Services Pillar documentation.

The Oracle Cloud Infrastructure Block Volume service lets you dynamically provision and manage block storage volumes. You can create, attach, connect and move volumes as needed to meet your storage and application requirements. The Block Volume service uses NVMe-based storage infrastructure, and is designed for consistency. You just need to provision the

capacity needed and performance scales linearly per GB volume size up to the service maximums. The following table describes the performance characteristics of the service.

Metric	Characteristic
Volume Size	50 GB to 32 TB, in 1 GB increments
IOPS	60 IOPS/GB , up to 25,000 IOPS
Throughput	480 KBPS/GB, up to 320 MBPS
Latency	Sub-millisecond latencies.
Per-instance Limits	32 attachments per instance, up to 1 PB. Up to 620K or more IOPS, near line rate throughout.

The following table lists the Block Volume service's throughput and IOPS performance numbers based on volume size. If you're trying to achieve certain performance targets you can provision a minimum volume size using this table as a reference.

Volume Size	Max Throughput (1 MB block size)	Max Throughput (8 KB block size)	Max IOPS (4 KB block size)
50 GB	24 MB/s	24 MB/s	3000
100 GB	48 MB/s	48 MB/s	6000
200 GB	96 MB/s	96 MB/s	12,000
300 GB	144 MB/s	144 MB/s	18,000
400 GB	192 MB/s	192 MB/s	24,000
500 GB	240 MB/s	200 MB/s	25,000
700 GB - 32 TB	320 MB/s	200 MB/s	25,000



Note

Performance Notes for Instance Types

- The throughput performance results are for bare metal instances. Throughput performance on VM instances is dependent on the network bandwidth that is available to the instance, and further limited by that bandwidth for the volume. For details about the network bandwidth available for VM shapes, see the **Network Bandwidth** column in the VM Shapes table.
- IOPS performance is independent of the instance type or shape, so is applicable to all bare metal and VM shapes, for iSCSI attached volumes. For VM shapes with paravirtualized attached volumes, see Paravirtualized Attachment Performance.
- Latency performance is independent of the instance shape or volume size, and is always sub-millisecond at the 95th percentile.
- There is no oversubscription of resources in Oracle Cloud Infrastructure to ensure stable and predictable throughput, IOPS, and latency performance.



Note

Testing Note for Windows Instances

Windows Defender Advanced Threat Protection (Windows Defender ATP) is enabled by default on all Oracle-provided Windows images. This tool has a significant negative impact on disk I/O performance. The IOPS performance characteristics described in this topic are valid for Windows bare metal instances with Windows Defender ATP disabled for disk I/O. Customers must carefully consider the security implications of disabling Windows Defender ATP. See Windows Defender Advanced Threat Protection.



Note

Paravirtualized Attachment Performance

The IOPS performance characteristics described in this topic are for volumes with iSCSI attachments. Block Volume performance SLA for IOPS per volume and IOPS per instance applies to iSCSI volume attachments only, not to paravirtualized attachments.

Paravirtualized attachments simplify the process of configuring your block storage by removing the extra commands needed before accessing a volume. However, due to the overhead of virtualization, this reduces the maximum IOPS performance for larger block volumes. If storage IOPS performance is of paramount importance for your workloads, you can continue to experience the guaranteed performance Oracle Cloud Infrastructure Block Volume offers by using iSCSI attachments.

For more information about FIO command samples you can use for performance testing see Sample FIO Commands for Block Volume Performance Tests on Linux-based Instances.

Testing Methodology and Performance



Warning

- Before running any tests, protect your data by making a backup of your data and operating system environment to prevent any data loss.
- Do not run FIO tests directly against a device that is already in use, such as /dev/sdX. If it is in use as a formatted disk and there is data on it, running FIO with a write workload (readwrite, randrw, write, trimwrite) will overwrite the data on the disk, and cause data corruption. Run FIO only on unformatted raw devices that are not in use.

This section describes the setup of the test environments, the methodology, and the observed performance. Some of the sample volume sizes tested were:

- 50 GB volume - 3,000 IOPS @ 4K
- 1 TB volume - 25,000 IOPS @ 4K
- Host maximum, Ashburn (IAD) region, twenty 1 TB volumes - 400,000 IOPS @ 4K

These tests used a wide range of volume sizes and the most common read and write patterns and were generated with the Gartner Cloud Harmony test suite. To show the throughput performance limits, 256k or larger block sizes should be used. For most environments, 4K, 8K, or 16K blocks are common depending on the application workload, and these are used specifically for IOPS measurements.

In the observed performance images in this section, the X axis represents the volume size tested, ranging from 4KB to 1MB. The Y axis represents the IOPS delivered. The Z axis represents the read/write mix tested, ranging from 100% read to 100% write.



Note

Performance Notes for Instance Types

- The throughput performance results are for bare metal instances. Throughput performance on VM instances is dependent on the network bandwidth that is available to the instance, and further limited by that bandwidth for the volume. For details about the network bandwidth available for VM shapes, see the Network Bandwidth column in the VM Shapes table.
- IOPS performance is independent of the instance type or shape, so is applicable to all bare metal and VM shapes, for iSCSI attached volumes. For VM shapes with paravirtualized attached volumes, see Paravirtualized Attachment Performance.

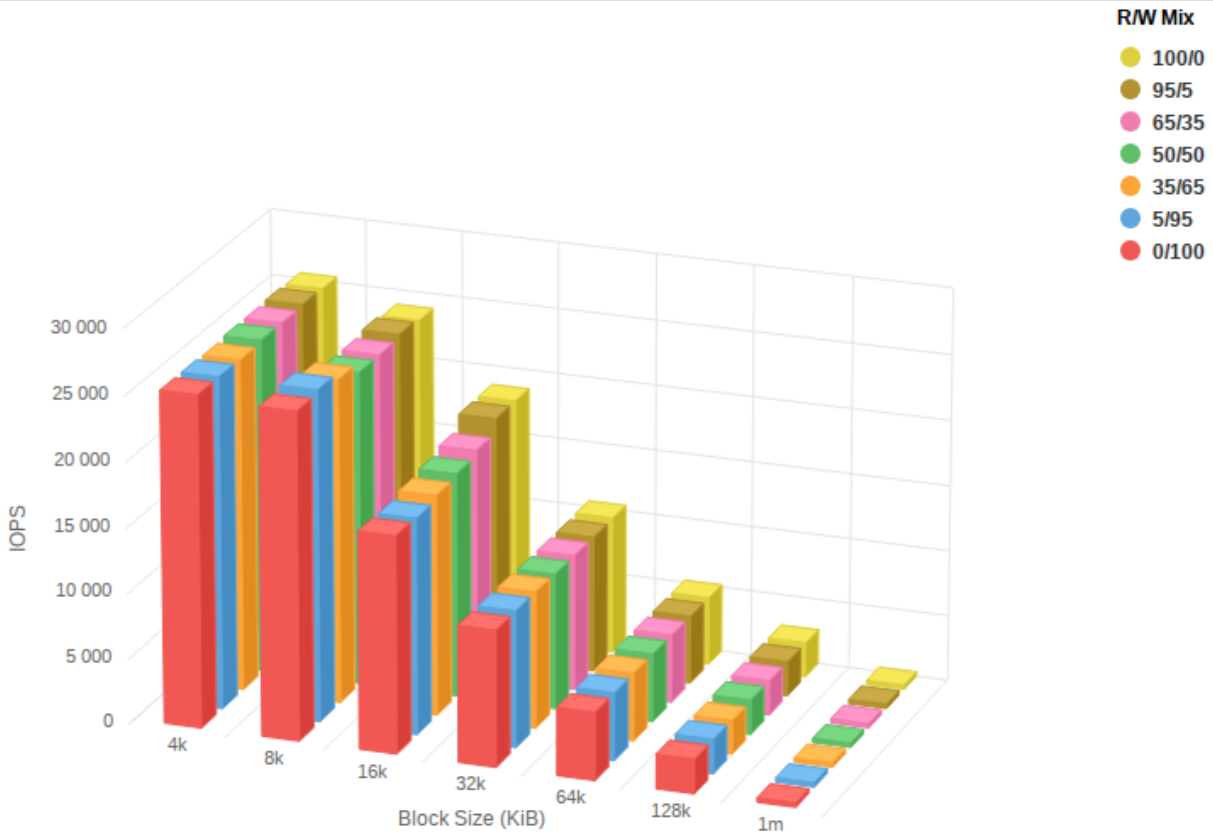
1 TB Block Volume

A 1 TB volume was mounted to a bare metal instance running in the Phoenix region. The instance shape was dense, workload was direct I/O with 10GB working set. The following command was run for the Gartner Cloud Harmony test suite:

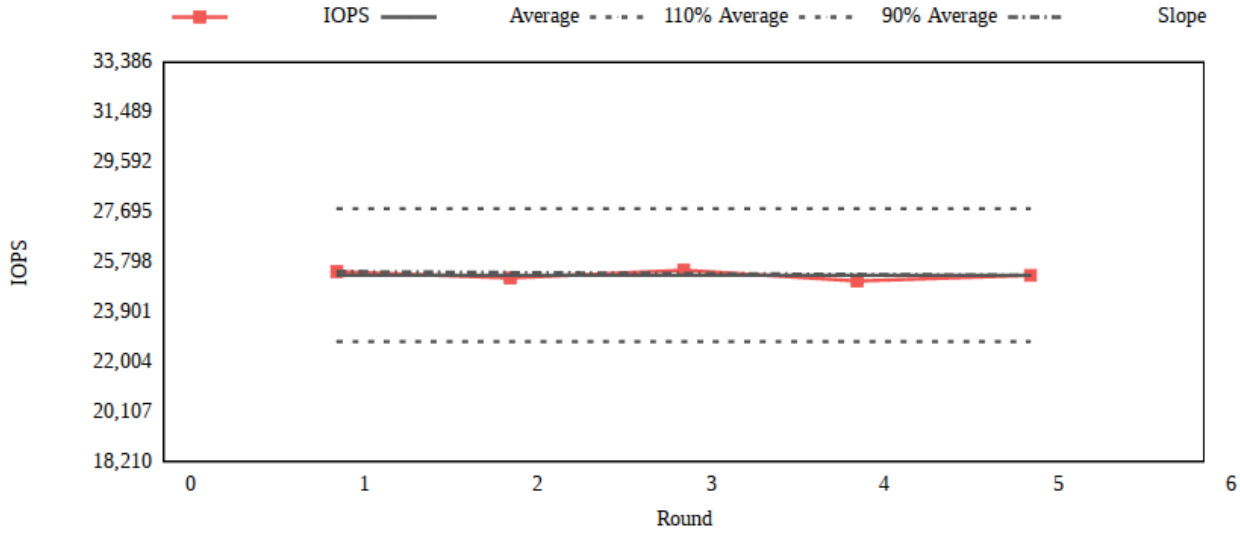
```
~/block-storage/run.sh --nopurge --noprecondition --fio_direct\=1 --fio_size=10g --target /dev/sdb --
test iops --skip_blocksize 512b
```

The results showed that for 1 TB, the bandwidth limit for the larger block size test occurs at 320MBS.

The following images show the observed performance for 1 TB:



Plot A.6 - IOPS -All RW Mix & BS - 3D Columns



Steady State Determination Data

Average IOPS:			25292
Allowed Maximum Data Excursion:	5058.497	Measured Maximum Data Excursion:	410.095321
Allowed Maximum Slope Excursion:	2529.2485	Measured Maximum Slope Excursion:	159.9
Least Squares Linear Fit Formula:			-39.978 * R + 25412.418

Plot A.3 - IOPS Steady State Measurement Window - RND/4KiB

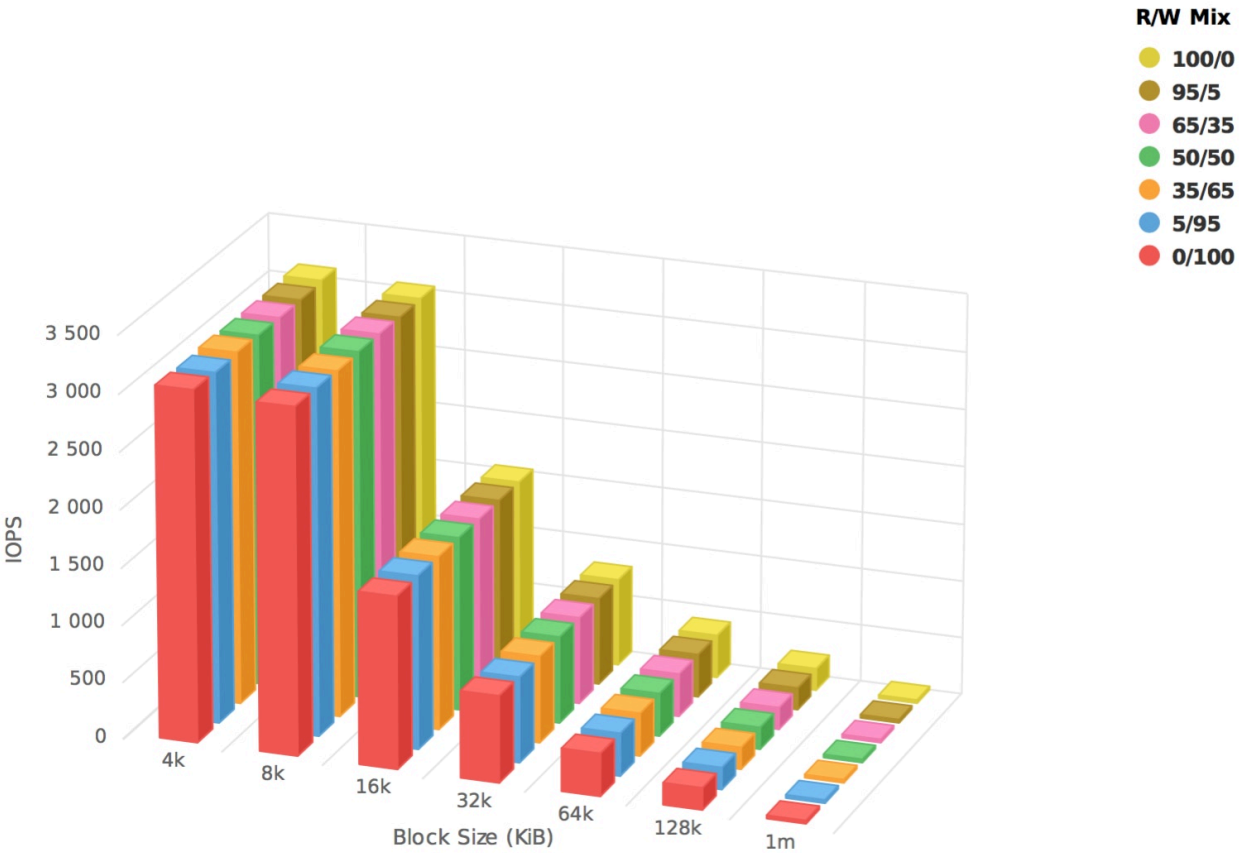
50 GB Block Volume

A 50 GB volume was mounted to a bare metal instance running in the Phoenix region. The instance shape was dense, workload was direct I/O with 10GB working set. The following command was run for the Gartner Cloud Harmony test suite:

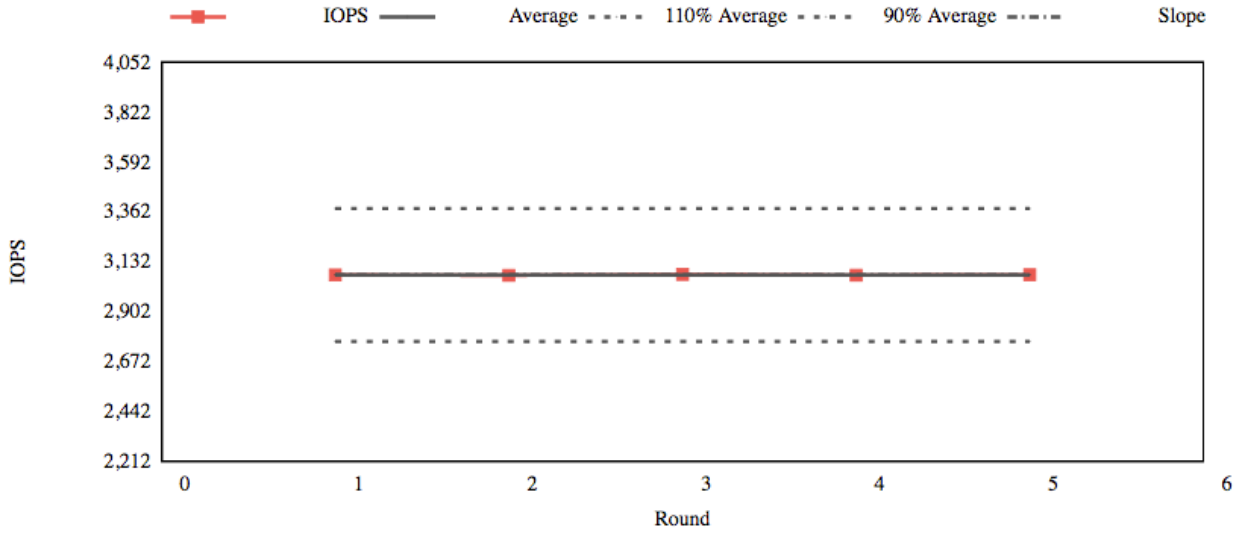
```
~/block-storage/run.sh --nopurge --noprecondition --fio_direct=1 --fio_size=10g --target /dev/sdb --test
iops --skip_blocksize 512b
```

The results showed that for the 50 GB volume, the bandwidth limit is confirmed as 24,000 KBPS for the larger block size tests (256 KB or larger block sizes), and the maximum of 3,000 IOPS at 4K block size is delivered. For small volumes, a 4K block size is common.

The following images show the observed performance for 50 GB:



Plot A.6 - IOPS -All RW Mix & BS - 3D Columns



Steady State Determination Data

Average IOPS:			3073
Allowed Maximum Data Excursion:	614.6282	Measured Maximum Data Excursion:	4.365131
Allowed Maximum Slope Excursion:	307.3141	Measured Maximum Slope Excursion:	1.6
Least Squares Linear Fit Formula:			0.4 * R + 3071.94

Plot A.3 - IOPS Steady State Measurement Window - RND/4KiB

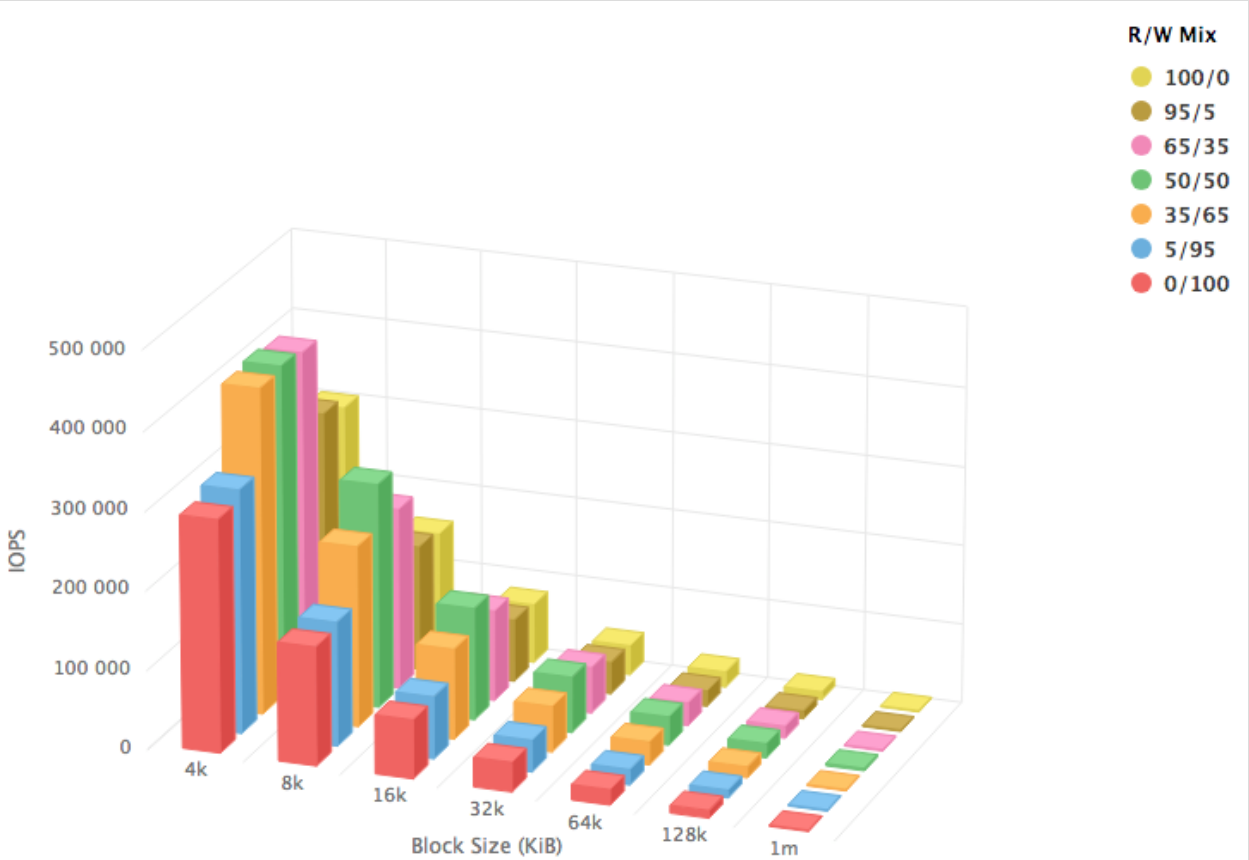
Host Maximum - Twenty 1 TB Volumes

Twenty 1 TB volumes were mounted to a bare metal instance running in the Ashburn region. The instance shape was dense, workload was direct I/O with 10GB working set. The following command was run for the Gartner Cloud Harmony test suite:

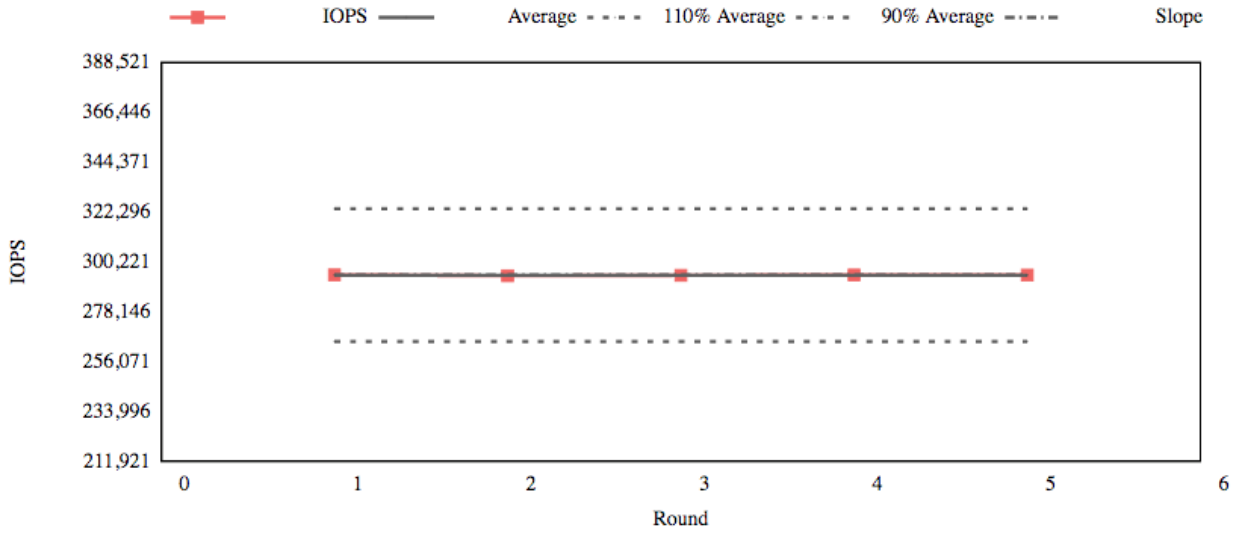
```
~/block-storage/run.sh --nopurge --noprecondition --fio_direct=1 --fio_size=10g --target
/dev/sdb,/dev/sdc,/dev/sdd,/dev/sde,/dev/sdf,/dev/sdg,/dev/sdh,/dev/sdi,/dev/sdj,/dev/sdk,/dev/sdl,/dev
/sdm,/dev/sdn,/dev/sdo,/dev/sdp,/dev/sdq,/dev/sdr,/dev/sds,/dev/sdt,/dev/sdu --test iops --skip_
blocksize 512b
```

The results showed that for the host maximum test of twenty 1 TB volumes, the average is 2.1GBPS, and 400,000 IOPS to the host for the 50/50 read/write pattern.

The following images show the observed performance for 50 GB:



Plot A.6 - IOPS -All RW Mix & BS - 3D Columns



Steady State Determination Data

Average IOPS:			294335
Allowed Maximum Data Excursion:	58867.0562	Measured Maximum Data Excursion:	338.11226299999
Allowed Maximum Slope Excursion:	29433.5281	Measured Maximum Slope Excursion:	58
Least Squares Linear Fit Formula:			14.492 * R + 294291.803

Plot A.3 - IOPS Steady State Measurement Window - RND/4KiB

Sample FIO Commands for Block Volume Performance Tests on Linux-based Instances

This topic describes sample FIO commands you can use to run performance tests for the Oracle Cloud Infrastructure Block Volume service on instances created from Linux-based images.

Installing FIO

To install and configure FIO on your instances with Linux-based operating systems run the commands applicable to the operating system version for your instance.

Oracle Linux and CentOS

Run the following command to install and configure FIO for your Oracle Linux CentOS systems:

```
sudo yum install fio -y
```

This applies to Oracle Linux 6.x, Oracle Linux 7.x, CentOS 6.x, and CentOS 7.x.

Ubuntu

Run the following commands to install and configure FIO for your Ubuntu systems:

```
sudo apt-get update && sudo apt-get install fio -y
```

This applies to Ubuntu 14.04, 16.04, 18.04, and Ubuntu Minimal 16.04, 18.04.

FIO Commands

IOPS Performance Tests

Use the following FIO example commands to test IOPS performance. You can run the commands directly or create a job file with the command and then run the job file.

Test random reads

Run the following command directly to test random reads:

```
sudo fio --filename=device name --direct=1 --rw=randread --bs=4k --ioengine=libaio --iodepth=256 --
runtime=120 --numjobs=4 --time_based --group_reporting --name=iops-test-job --eta-newline=1 --readonly
```

In some cases you may see more consistent results if you use a job file instead of running the command directly. Use the following steps for this approach.

1. Create a job file, fiorandomread.fio, with the following:

   ```
   [global]
   bs=4K
   iodepth=256
   ```

```
direct=1
ioengine=libaio
group_reporting
time_based
runtime=120
numjobs=4
name=raw-randread
rw=randread

[job1]
filename=device name
```

2. Run the job using the following command:

```
fio randomread.fio
```

Test file random read/writes

Run the following command against the mount point to test file read/writes:

```
sudo fio --filename=/custom mount point/file --size=500GB --direct=1 --rw=randrw --bs=4k --
ioengine=libaio --iodepth=256 --runtime=120 --numjobs=4 --time_based --group_reporting --name=iops-test-
job --eta-newline=1
```

Add both the read IOPS and the write IOPS returned.

Test random read/writes



Warning

Do not run FIO tests with a write workload (`readwrite`, `randrw`, `write`, `trimwrite`) directly against a device that is in use.

Run the following command to test random read/writes:

```
sudo fio --filename=device name --direct=1 --rw=randrw --bs=4k --ioengine=libaio --iodepth=256 --
runtime=120 --numjobs=4 --time_based --group_reporting --name=iops-test-job --eta-newline=1
```

Add both the read IOPS and the write IOPS returned.

In some cases you may see more consistent results if you use a job file instead of running the command directly. Use the following steps for this approach.

1. Create a job file, fiorandomreadwrite.fio, with the following:

   ```
   [global]
   bs=4K
   iodepth=256
   direct=1
   ioengine=libaio
   group_reporting
   time_based
   runtime=120
   numjobs=4
   name=raw-randreadwrite
   rw=randrw

   [job1]
   filename=device name
   ```

2. Run the job using the following command:

   ```
   fio randomreadwrite.fio
   ```

Test sequential reads

For workloads that enable you to take advantage of sequential access patterns, such as database workloads, you can confirm performance for this pattern by testing sequential reads.

Run the following command to test sequential reads:

```
sudo fio --filename=device name --direct=1 --rw=read --bs=4k --ioengine=libaio --iodepth=256 --
runtime=120 --numjobs=4 --time_based --group_reporting --name=iops-test-job --eta-newline=1 --readonly
```

In some cases you may see more consistent results if you use a job file instead of running the command directly. Use the following instructions for this approach:

1. Create a job file, fioread.fio, with the following:

```
[global]
bs=4K
iodepth=256
direct=1
ioengine=libaio
group_reporting
time_based
runtime=120
numjobs=4
name=raw-read
rw=read

[job1]
filename=device name
```

2. Run the job using the following command:

```
fio read.fio
```

THROUGHPUT PERFORMANCE TESTS

Use the following FIO example commands to test throughput performance.

Test random reads

Run the following command to test random reads:

```
sudo fio --filename=device name --direct=1 --rw=randread --bs=64k --ioengine=libaio --iodepth=64 --
runtime=120 –numjobs=4 --time_based --group_reporting --name=throughput-test-job --eta-newline=1 --
readonly
```

In some cases you may see more consistent results if you use a job file instead of running the command directly. Use the following steps for this approach.

1. Create a job file, fiorandomread.fio, with the following:

```
[global]
bs=64K
iodepth=64
direct=1
ioengine=libaio
group_reporting
time_based
runtime=120
numjobs=4
name=raw-randread
rw=randread

[job1]
filename=device name
```

2. Run the job using the following command:

```
fio randomread.fio
```

Test file random read/writes

Run the following command against the mount point to test file read/writes:

```
sudo fio --filename=/custom mount point/file --size=500GB --direct=1 --rw=randrw --bs=64k --
ioengine=libaio --iodepth=64 --runtime=120 --numjobs=4 --time_based --group_reporting --name=throughput-
test-job --eta-newline=1
```

Add both the read MBPs and the write MBPs returned.

Test random read/writes



Warning

Do not run FIO tests with a write workload (`readwrite`, `randrw`, `write`, `trimwrite`) directly against a device



that is in use.

Run the following command to test random read/writes:

```
sudo fio --filename=device name --direct=1 --rw=randrw --bs=264k --ioengine=libaio --iodepth=64 --
runtime=120 --numjobs=4 --time_based --group_reporting --name=throughput-test-job --eta-newline=1
```

Add both the read MBPs and the write MBPs returned.

In some cases you may see more consistent results if you use a job file instead of running the command directly. Use the following steps for this approach.

1. Create a job file, fiorandomread.fio, with the following:

```
[global]
bs=264K
iodepth=64
direct=1
ioengine=libaio
group_reporting
time_based
runtime=120
numjobs=4
name=raw-randreadwrite
rw=randrw

[job1]
filename=device name
```

2. Run the job using the following command:

```
fio randomreadwrite.fio
```

Test sequential reads

For workloads that enable you to take advantage of sequential access patterns, such as database workloads, you can confirm performance for this pattern by testing sequential reads.

Run the following command to test sequential reads:

```
sudo fio --filename=device name --direct=1 --rw=read --bs=64k --ioengine=libaio --iodepth=64 --
runtime=120 —numjobs=4 --time_based --group_reporting --name=throughput-test-job --eta-newline=1 --
readonly
```

In some cases you may see more consistent results if you use a job file instead of running the command directly. Use the following steps for this approach.

1. Create a job file, fioread.fio, with the following:

   ```
   [global]
   bs=64K
   iodepth=64
   direct=1
   ioengine=libaio
   group_reporting
   time_based
   runtime=120
   numjobs=4
   name=raw-read
   rw=read

   [job1]
   filename=device name
   ```

2. Run the job using the following command:

   ```
   fio read.fio
   ```

Block Volume Metrics

You can monitor the health, capacity, and performance of your block volumes and boot volumes by using metrics, alarms, and notifications.

This topic describes the metrics emitted by the metric namespace `oci_blockstore` (the Block Volume service).

Resources: Block volumes and boot volumes.

Overview of Metrics for an Instance and Its Storage Devices

If you're not already familiar with the different types of metrics available for an instance and its storage and network devices, see Compute Metrics.

Available Metrics: oci_blockstore

The Block Volume service metrics help you measure volume operations and throughput related to Compute instances.

The metrics listed in the following table are automatically available for any block volume or boot volume, regardless of whether the attached instance has monitoring enabled. You do not need to enable monitoring on the volumes to get these metrics.

You also can use the Monitoring service to create custom queries.

Each metric includes the following dimensions:

ATTACHMENTID

The OCID of the volume attachment.

RESOURCEID

The OCID of the volume.

Metric	Metric Display Name	Unit	Description	Dimensions
`VolumeReadThroughput`*	**Volume Read Throughput**	bytes	Read throughput. Expressed as bytes read per interval.	`attachmentId` `resourceId`
`VolumeWriteThroughput`*	**Volume Write Throughput**	bytes	Write throughput. Expressed as bytes written per interval.	
`VolumeReadOps`*	**Volume Read Operations**	reads	Activity level from I/O reads. Expressed as reads per interval.	
`VolumeWriteOps`*	**Volume Write Operations**	writes	Activity level from I/O writes. Expressed as writes per interval.	

* The Compute service separately reports network-related metrics *as measured on the instance itself and aggregated across all the attached volumes*. Those metrics are available in the `oci_computeagent` metric namespace. For more information, see Compute Metrics.

Using the Console

To view default metric charts for a single volume

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
2. Click the instance to view its details.

3. Click either **Attached Block Volumes** or **Boot Volume** to view the volume you're interested in.
4. Click the volume to view its details.
5. Under **Resources**, click **Metrics**.

For more information about monitoring metrics and using alarms, see Monitoring Overview. For information about notifications for alarms, see Notifications Overview.

To view default metric charts for multiple volumes

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Service Metrics**.
2. For **Compartment**, select the compartment that contains the volumes you're interested in.
3. For **Metric Namespace**, select **oci_blockstore**.
   The **Service Metrics** page dynamically updates the page to show charts for each metric that is emitted by the selected metric namespace.

For more information about monitoring metrics and using alarms, see Monitoring Overview. For information about notifications for alarms, see Notifications Overview.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following APIs for monitoring:

- Monitoring API for metrics and alarms
- Notifications API for notifications (used with alarms)

CHAPTER 6 Compute

This chapter explains how to launch, access, rename, and terminate compute instances.

Overview of the Compute Service

Oracle Cloud Infrastructure Compute lets you provision and manage compute hosts, known as instances. You can launch instances as needed to meet your compute and application requirements. After you launch an instance, you can access it securely from your computer, restart it, attach and detach volumes, and terminate it when you're done with it. Any changes made to the instance's local drives are lost when you terminate it. Any saved changes to volumes attached to the instance are retained.

Oracle Cloud Infrastructure offers both bare metal and virtual machine instances:

- **Bare Metal:** A bare metal compute instance gives you dedicated physical server access for highest performance and strong isolation.
- **Virtual Machine:** A virtual machine (VM) is an independent computing environment that runs on top of physical bare metal hardware. The virtualization makes it possible to run multiple VMs that are isolated from each other. VMs are ideal for running applications that do not require the performance and resources (CPU, memory, network bandwidth, storage) of an entire physical machine.

  An Oracle Cloud Infrastructure VM compute instance runs on the same hardware as a bare metal instance, leveraging the same cloud-optimized hardware, firmware, software stack, and networking infrastructure.

Be sure to review Best Practices for Your Compute Instance for important information about working with your Oracle Cloud Infrastructure Compute instance.

Oracle Cloud Infrastructure uses Oracle Ksplice to apply important security and other critical kernel updates to the hypervisor hosts without a reboot.

Instance Types

When you create a Compute instance, you can select the most appropriate type of instance for your applications based on characteristics such as the number of CPUs, amount of memory, and network resources. Oracle Cloud Infrastructure offers a variety of shapes that are designed to meet a range of compute and application requirements:

- **Standard shapes:** Designed for general purpose workloads and suitable for a wide range of applications and use cases. Standard shapes provide a balance of cores, memory, and network resources. Standard shapes are available with Intel or AMD processors.
- **DenseIO shapes:** Designed for large databases, big data workloads, and applications that require high-performance local storage. DenseIO shapes include locally-attached NVMe-based SSDs.
- **GPU shapes:** Designed for hardware-accelerated workloads. GPU shapes include Intel CPUs and NVIDIA graphics processors.
- **High performance computing (HPC) shapes:** Designed for high-performance computing workloads that require high frequency processor cores and cluster networking for massively parallel HPC workloads. HPC shapes are available for bare metal instances only.

For more information about the available bare metal and VM shapes, see Compute Shapes, Bare Metal Instance Types, Virtual Machine Shapes, and Accelerated GPU Instance Types.

Components for Launching Instances

The components required to launch an instance are:

AVAILABILITY DOMAIN

The Oracle Cloud Infrastructure data center within your geographical region that hosts cloud resources, including your instances. You can place instances in the same or different availability domains, depending on your performance and redundancy requirements. For more information, see Regions and Availability Domains.

VIRTUAL CLOUD NETWORK

A virtual version of a traditional network—including subnets, route tables, and gateways—on which your instance runs. At least one cloud network has to be set up before you launch instances. For information about setting up cloud networks, see Overview of Networking.

KEY PAIR (FOR LINUX INSTANCES)

A security mechanism required for Secure Shell (SSH) access to an instance. Before you launch an instance, you'll need at least one key pair. For more information, see Managing Key Pairs on Linux Instances.

TAGS

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

PASSWORD (FOR WINDOWS INSTANCES)

A security mechanism required to access an instance that uses an Oracle-provided Windows image. The first time you launch an instance using a Windows image, Oracle Cloud Infrastructure will generate an initial, one-time password that you can retrieve using the console or API. This password must be changed after you initially log on.

IMAGE

A template of a virtual hard drive that determines the operating system and other software for an instance. For details about Oracle Cloud Infrastructure platform images, see Oracle-Provided Images. You can also launch instances from:

- Images published by Oracle partners from the Partner Image catalog.
- Pre-built Oracle enterprise images and solutions enabled for Oracle Cloud Infrastructure
- Custom images, including bring your own image scenarios.
- Boot Volumes.

SHAPE

A template that determines the number of CPUs, amount of memory, and other resources allocated to a newly created instance. You choose the most appropriate shape when you launch an instance. See Compute Shapes for a list of available bare metal and VM shapes.

You can optionally attach volumes to an instance. For more information, see Overview of Block Volume.

Resource Identifiers

Most types of Oracle Cloud Infrastructure resources have a unique, Oracle-assigned identifier called an Oracle Cloud ID (OCID). For information about the OCID format and other ways to identify your resources, see Resource Identifiers.

Ways to Access Oracle Cloud Infrastructure

You can access Oracle Cloud Infrastructure using the Console (a browser-based interface) or the REST API. Instructions for the Console and API are included in topics throughout this guide. For a list of available SDKs, see Software Development Kits and Command Line Interface.

To access the Console, you must use a supported browser. Oracle Cloud Infrastructure supports the latest desktop versions of Google Chrome, Microsoft Edge, Internet Explorer 11, Safari, Firefox, and Firefox ESR. Note that private browsing mode is not supported for Firefox, Internet Explorer, or Edge. Mobile browsers are not supported.

For general information about using the API, see REST APIs.

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For

example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

Limits on Compute Resources

See Service Limits for a list of applicable limits and instructions for requesting a limit increase.

Additional limits include:

- To attach a volume to an instance, both the instance and volume must be within the same availability domain.
- Many Compute operations are subject to throttling.

Metadata Key Limits

Custom metadata keys (any key you define that is not `ssh_authorized_keys` or `user_data`) have the following limits:

- Max number of metadata keys: 128
- Max size of key name: 255 characters
- Max size of key value: 255 characters

`ssh_authorized_keys` is a special key that does not have these limits, but its value is validated to conform to a public key in the OpenSSH format.

`user_data` has a maximum size of 16KB. For Linux instances with cloud-init configured, you can populate the `user_data` field with a Base64-encoded string of cloud-init user data. For

more information on formats that cloud-init accepts, see cloud-init formats. On Windows instances, the `user_data` field can be provided but isn't used by Oracle-provided images.

Best Practices for Your Compute Instance

Oracle Cloud Infrastructure Compute provides bare metal compute capacity that delivers performance, flexibility, and control without compromise. It is powered by Oracle's next generation, internet-scale infrastructure designed to help you develop and run your most demanding applications and workloads in the cloud.

You can provision compute capacity through an easy-to-use web console or an API. The bare metal compute instance, once provisioned, provides you with access to the host. This gives you complete control of your instance.

While you have full management authority for your instance, Oracle recommends a variety of best practices to ensure system availability and top performance.

IP Addresses Reserved for Use by Oracle

Certain IP addresses are reserved for Oracle Cloud Infrastructure use and may not be used in your address numbering scheme.

169.254.0.0/16

These addresses are used for iSCSI connections to the boot and block volumes, instance metadata, and other services.

Three IP Addresses in Each Subnet

These addresses consist of:

- The first IP address in the CIDR (the network address)
- The last IP address in the CIDR (the broadcast address)
- The first host address in the CIDR (the subnet default gateway address)

For example, in a subnet with CIDR 192.168.0.0/24, these addresses are reserved:

- 192.168.0.0 (the network address)
- 192.168.0.255 (the broadcast address)
- 192.168.0.1 (the subnet default gateway address)

The remaining addresses in the CIDR (192.168.0.2 to 192.168.0.254) are available for use.

Essential Firewall Rules



Warning

Windows 2008 Server R2 images do not support restricting certain firewall rules for local principals, such as "Administrators", so any authenticated user on an instance can make outgoing connections to the iSCSI network endpoints (169.254.0.2:3260, 169.254.2.0/24:3260) that serve the instance's boot and block volumes.

All Oracle-provided images include rules that allow only "root" on Linux instances or "Administrators" on Windows Server 2012 R2 and Windows Server 2016 instances to make outgoing connections to the iSCSI network endpoints (169.254.0.2:3260, 169.254.2.0/24:3260) that serve the instance's boot and block volumes.

- Oracle recommends that you do not reconfigure the firewall on your instance to remove these rules. Removing these rules allows non-root users or non-administrators to access the instance’s boot disk volume.
- Oracle recommends that you do not create custom images without these rules unless you understand the security risks.
- Running Uncomplicated Firewall (UFW) on Ubuntu images may cause issues with these rules, so Oracle recommends that you do not enable UFW on your instances. See Ubuntu Instance fails to reboot after enabling Uncomplicated Firewall (UFW) for more information.

System Resilience

Oracle Cloud Infrastructure runs on Oracle's high-quality Sun servers. However, any hardware can experience a failure. Follow industry-wide hardware failure best practices to ensure the resilience of your solution. Some best practices include:

- Design your system with redundant compute nodes in different availability domains to support fail-over capability.
- Create a custom image of your system drive each time you change the image.
- Back up your data drives, or sync to spare drives, regularly.

If you experience a hardware failure and have followed these practices, you can terminate the failed instance, launch your custom image to create a new instance, and then apply the backup data.

Uninterrupted Access to the Instance

Make sure to keep the DHCP client running so you can always access the instance. If you stop the DHCP client manually or disable NetworkManager (which stops the DHCP client on Linux

instances), the instance can't renew its DHCP lease and will become inaccessible when the lease expires (typically within 24 hours). Do not disable NetworkManager unless you use another method to ensure renewal of the lease.

Stopping the DHCP client might remove the host route table when the lease expires. Also, loss of network connectivity to your iSCSI connections might result in loss of the boot drive.

User Access

If you created your instance using an Oracle-provided Linux image, you can use SSH to access your instance from a remote host as the `opc` user. After logging in, you can add users on your instance.

If you do not want to share SSH keys, you can create additional SSH-enabled users.

If you created your instance using an Oracle-provided Windows image, you can access your instance using a Remote Desktop client as the `opc` user. After logging in, you can add users on your instance.

For more information about user access, see Adding Users on an Instance.

NTP Service

Oracle Cloud Infrastructure offers a fully managed, secure, and highly available NTP service that you can use to set the date and time of your Compute and Database instances from within your virtual cloud network (VCN). Oracle recommends that you configure your instances to use the Oracle Cloud Infrastructure NTP service. For information about how to configure instances to use this service, see Configuring the Oracle Cloud Infrastructure NTP Service for an Instance.

Fault Domains

A fault domain is a grouping of hardware and infrastructure that is distinct from other fault domains in the same availability domain. Each availability domain has three fault domains. By properly leveraging fault domains you can increase the availability of applications running on Oracle Cloud Infrastructure. See Fault Domains for more information.

Your application's architecture will determine whether you should separate or group instances using fault domains.

Scenario 1: Highly Available Application Architecture

In this scenario you have a highly available application, for example you have two web servers and a clustered database. In this scenario you should group one web server and one database node in one fault domain and the other half of each pair in another fault domain. This ensures that a failure of any one fault domain does not result in an outage for your application.

Scenario 2: Single Web Server and Database Instance Architecture

In this scenario your application architecture is not highly available, for example you have one web server and one database instance. In this scenario both the web server and the database instance must be placed in the same fault domain. This ensures that your application will only be impacted by the failure of that single fault domain.

Customer-Managed Virtual Machine (VM) Maintenance

Today when an underlying infrastructure component needs to undergo maintenance, we notify you in advance of the planned maintenance downtime. To avoid this planned downtime, you have the options to reboot, or stop and restart your instances prior to the scheduled maintenance. This makes it easy for you to control your instance downtime during the notification period. The reboot, or stop and restart of a VM instance during the notification period is different than a normal reboot. The reboot, or stop and start workflow will stop your instance on the existing VM host which needs maintenance and restarts it on a healthy VM host. If you choose not to reboot during the notification period, then Oracle Cloud Infrastructure will reboot your VM Instance before proceeding with the planned infrastructure maintenance. Customer-managed VM maintenance is currently supported only on standard instance shapes running Linux-based operating systems. This includes Oracle-provided platform images or custom images imported from outside of Oracle Cloud Infrastructure. For information on rebooting or restarting your instance prior to planned maintenance, see Stopping and Starting an Instance.

Configuring the Oracle Cloud Infrastructure NTP Service for an Instance

Oracle Cloud Infrastructure offers a fully managed, secure, and highly available NTP service that you can use to set the date and time of your Compute and Database instances from within your virtual cloud network (VCN). The Oracle Cloud Infrastructure NTP service uses redundant Stratum 1 devices in every availability domain. The Stratum 1 devices are synchronized to dedicated Stratum 2 devices that every host synchronizes against. The service is available in every region.

This topic describes how to configure your Compute instances to use this NTP service.

You can also choose to configure your instances to use a public NTP service or use FastConnect to leverage an on-premises NTP service.

Oracle Linux 6.x

Use the following steps to configure your Oracle Linux 6.x instances to use the Oracle Cloud Infrastructure NTP service.

1. Configure IPtables to allow connections to the Oracle Cloud Infrastructure NTP service, using the following commands:

   ```
   sudo iptables -I BareMetalInstanceServices 8 -d 169.254.169.254/32 -p udp -m udp --dport 123 -m comment --comment "Allow access to OCI local NTP service" -j ACCEPT
   ```

   ```
   sudo service iptables save
   ```

2. Install the NTP service with the following command:

   ```
   sudo yum install ntp
   ```

3. Set the date of your instance with the following command:

   ```
   sudo ntpdate 169.254.169.254
   ```

4. Configure the instance to use the Oracle Cloud Infrastructure NTP service for iburst. To configure, modify the `/etc/ntp.conf` file as follows:

a. In the `server` section, comment out the lines specifying the RHEL servers:

```
#server 0.rhel.pool.ntp.org iburst
#server 1.rhel.pool.ntp.org iburst
#server 2.rhel.pool.ntp.org iburst
#server 3.rhel.pool.ntp.org iburst
```

b. Add an entry for the Oracle Cloud Infrastructure NTP server:

```
server 169.254.169.254 iburst
```

The modified `server` section now contains the following:

```
# Please consider joining the pool (http://www.pool.ntp.org/join.html).
#server 0.rhel.pool.ntp.org iburst
#server 1.rhel.pool.ntp.org iburst
#server 2.rhel.pool.ntp.org iburst
#server 3.rhel.pool.ntp.org iburst
server 169.254.169.254 iburst
```

5. Set the NTP service to launch automatically when the instance boots with the following command:

```
sudo chkconfig ntpd on
```

6. Start the NTP service with the following command:

```
sudo /etc/init.d/ntpd start
```

7. Confirm that the NTP service is configured correctly with the following command:

```
ntpq -p
```

The output will be similar to the following:

```
remote           refid      st t when poll reach   delay   offset  jitter
==============================================================================
169.254.169.254 192.168.32.3     2 u    2   64    1    0.338    0.278   0.187
```

Oracle Linux 7.x



Note

Oracle-provided Oracle 7.x and CentOS 7.x images released after February 2018 include the Chrony service by default, so you do not need to configure the Oracle Cloud Infrastructure NTP service for these instances.

Use the following steps to configure your Oracle Linux 7.x instances to use the Oracle Cloud Infrastructure NTP service.

1. Run commands in this section as root with the following command:

   ```
   sudo su -
   ```

2. Install the NTP service with the following command:

   ```
   yum -y install ntp
   ```

3. Change the firewall rules to allow inbound and outbound traffic with the Oracle Cloud Infrastructure NTP server, at 169.254.169.254, on UDP port 123 with the following command:

   ```
   awk -v n=13 -v s=' <passthrough ipv="ipv4">-A OUTPUT -d 169.254.169.254/32 -p udp -m udp --dport
   123 -m comment --comment "Allow access to OCI local NTP service" -j ACCEPT </passthrough>' 'NR ==
   n {print s} {print}' /etc/firewalld/direct.xml > tmp && mv tmp /etc/firewalld/direct.xml
   ```

   At the prompt:

   ```
   mv: overwrite '/etc/firewalld/direct.xml'?
   ```

   enter `y`

4. Restart the firewall with the following command:

   ```
   service firewalld restart
   ```

5. Set the date of your instance with the following command:

```
ntpdate 169.254.169.254
```

6. Configure the instance to use the Oracle Cloud Infrastructure NTP service for iburst. To configure, modify the `/etc/ntp.conf` file as follows:
   a. In the `server` section comment out the lines specifying the RHEL servers:

      ```
      #server 0.rhel.pool.ntp.org iburst
      #server 1.rhel.pool.ntp.org iburst
      #server 2.rhel.pool.ntp.org iburst
      #server 3.rhel.pool.ntp.org iburst
      ```

   b. Add an entry for the Oracle Cloud Infrastructure NTP service:

      ```
      server 169.254.169.254 iburst
      ```

   The modified `server` section should now contain the following:

   ```
   # Please consider joining the pool (http://www.pool.ntp.org/join.html).
   #server 0.rhel.pool.ntp.org iburst
   #server 1.rhel.pool.ntp.org iburst
   #server 2.rhel.pool.ntp.org iburst
   #server 3.rhel.pool.ntp.org iburst
   server 169.254.169.254 iburst
   ```

7. Start and enable the NTP service with the following commands:

   ```
   systemctl start ntpd
   systemctl enable ntpd
   ```

   You also need disable the chrony NTP client to ensure that the NTP service starts automatically after a reboot, using the following commands:

   ```
   systemctl stop chronyd
   systemctl disable chronyd
   ```

8. Confirm that the NTP service is configured correctly with the following command:

   ```
   ntpq -p
   ```

   The output will be similar to the following:

   ```
        remote           refid      st t when poll reach   delay   offset  jitter
   ==============================================================================
   ```

```
169.254.169.254 192.168.32.3     2 u    2   64    1    0.338    0.278   0.187
```

Windows Server 2016, Windows Server 2012 R2 and Windows Server 2008 R2

You can configure your Windows Server instances to use the Oracle Cloud Infrastructure NTP service by running the following commands in Windows Powershell as Administrator.

Windows 2012 and Windows 2016:

```
Set-ItemProperty -Path 'HKLM:\System\CurrentControlSet\Services\W32Time\Parameters' -Name 'Type' -Value
NTP -Type String
Set-ItemProperty -Path 'HKLM:\System\CurrentControlSet\Services\W32Time\Config' -Name 'AnnounceFlags' -
Value 5 -Type DWord
Set-ItemProperty -Path 'HKLM:\System\CurrentControlSet\Services\W32Time\TimeProviders\NtpServer' -Name
'Enabled' -Value 1 -Type DWord
Set-ItemProperty -Path 'HKLM:\System\CurrentControlSet\Services\W32Time\Parameters' -Name 'NtpServer' -
Value '169.254.169.254,0x9' -Type String
Set-ItemProperty -Path 'HKLM:\System\CurrentControlSet\Services\W32Time\TimeProviders\NtpClient' -Name
'SpecialPollInterval' -Value 900 -Type DWord
Set-ItemProperty -Path 'HKLM:\System\CurrentControlSet\Services\W32Time\Config' -Name
'MaxPosPhaseCorrection' -Value 1800 -Type DWord
Set-ItemProperty -Path 'HKLM:\System\CurrentControlSet\Services\W32Time\Config' -Name
'MaxNegPhaseCorrection' -Value 1800 -Type DWord
```

Windows 2008:

```
Set-ItemProperty -Path 'HKLM:\System\CurrentControlSet\Services\W32Time\Parameters' -Name 'Type' -Value
NTP -Type String
Set-ItemProperty -Path 'HKLM:\System\CurrentControlSet\Services\W32Time\Config' -Name 'AnnounceFlags' -
Value 5 -Type DWord
Set-ItemProperty -Path 'HKLM:\System\CurrentControlSet\Services\W32Time\TimeProviders\NtpServer' -Name
'Enabled' -Value 1 -Type DWord
Set-ItemProperty -Path 'HKLM:\System\CurrentControlSet\Services\W32Time\Parameters' -Name 'NtpServer' -
Value '169.254.169.254,0x9' -Type String
Set-ItemProperty -Path 'HKLM:\System\CurrentControlSet\Services\W32Time\TimeProviders\NtpClient' -Name
'SpecialPollInterval' -Value 900 -Type DWord
Set-ItemProperty -Path 'HKLM:\System\CurrentControlSet\Services\W32Time\Config' -Name
'MaxPosPhaseCorrection' -Value 1800 -Type DWord
Set-ItemProperty -Path 'HKLM:\System\CurrentControlSet\Services\W32Time\Config' -Name
'MaxNegPhaseCorrection' -Value 1800 -Type DWord
```

```
Set-ItemProperty -Path 'HKLM:\SOFTWARE\Microsoft\Windows\CurrentVersion\DateTime\Servers' -Name 0 -Value
"169.254.169.254"
Set-ItemProperty -Path 'HKLM:\SOFTWARE\Microsoft\Windows\CurrentVersion\DateTime\Servers' -Name "
(Default)" -Value "0"
```

Steps 1 - 7 below walk you though these registry changes, you can use these steps to manually edit the registry instead of using PowerShell. If you use the PowerShell commands, you can skip steps 1 - 7, and proceed with steps 8 and 9 to complete the process of configuring your Windows instance to use the Oracle Cloud Infrastructure NTP service.

1. Change the server type to NTP:
    a. From Registry Editor, navigate to:

    ```
    HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W32Time\Parameters\
    ```

    b. Click **Type**.
    c. Change the value to `NTP` and click **OK**.
2. Configure the Windows Time service to enable the `Timeserv_Announce_Yes` and `Reliable_Timeserv_Announce_Auto` flags.
   To configure, set the `AnnounceFlags` parameter to 5:
    a. From Registry Editor, navigate to:

    ```
    HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W32Time\Config\
    ```

    b. Click **AnnounceFlags**.
    c. Change the value to `5` and click **OK**.
3. Enable the NTP server:
    a. From Registry Editor, navigate to:

    ```
    HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W32Time\TimeProviders\NtpServer\
    ```

    b. Click **Enabled**.
    c. Change the value to `1` and click **OK**.
4. Set the time sources:

a. From Registry Editor, navigate to:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W32Time\Parameters\
```

b. Click **NtpServer**.

c. Change the value to `169.254.169.254,0x9` and click **OK**.

5. Set the poll interval:

a. From Registry Editor, navigate to:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W32Time\TimeProviders\NtpClient\
```

b. Click **SpecialPollInterval**.

c. Set the value to the interval that you want the time service to synchronize on. The value is in seconds. To set it for 15 minutes, set the value to `900`, and click **OK**.

6. Set the phase correction limit settings to restrict the time sample boundaries:

a. From Registry Editor, navigate to:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W32Time\Config\
```

b. Click **MaxPosPhaseCorrection**.

c. Set the value to the maximum time offset in the future for time samples. The value is in seconds. To set it for 30 minutes, set the value to `1800` and click **OK**.

d. Click **MaxNegPhaseCorrection**.

e. Set the value to the maximum time offset in the past for time samples. The value is in seconds. To set it for 30 minutes, set the value to `1800` and click **OK**.

7. For Windows 2008 only, you need to add the Oracle NTP service to the list of available servers and then configure the Oracle NTP service to be the active server:

a. From Registry Editor, navigate to:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\DateTime\Servers\
```

b. Right-click **Servers**, and select **New**, **String Value**.

c. Set the **Name** to `0`.

d. Set the **Value data** to `169.254.169.254`.

e. In the same path, find the string value named `(Default)` and double-click it to open the **Edit String** dialog.

f. Set the **Value data** to `0` and click **OK**.

8. Restart the time service by running the following command from a command prompt:

```
net stop w32time && net start w32time
```

9. Test the connection to the NTP service by running the following command from a command prompt:

```
w32tm /query /peers
```

The output will be similar to the following:

```
#Peer: 1

Peer: 169.254.169.254,0x9
State: Active
Time Remaining: 22.1901786s
Mode: 3 (Client)
Stratum: 0 (unspecified)
PeerPoll Interval: 10 (1024s)
HostPoll Interval: 10 (1024s)
```

After the time specified in the poll interval has elapsed, `State` will change from `Pending` to `Active`.

Protecting Data on NVMe Devices

Some instance shapes in Oracle Cloud Infrastructure include locally attached NVMe devices. These devices provide extremely low latency, high performance block storage that is ideal for big data, OLTP, and any other workload that can benefit from high-performance block storage.

Note that these devices are not protected in any way; they are individual devices locally installed on your instance. Oracle Cloud Infrastructure does not take images, back up, or use RAID or any other methods to protect the data on NVMe devices. It is your responsibility to protect and manage the durability the data on these devices.

Oracle Cloud Infrastructure offers high-performance remote block (iSCSI) LUNs that are redundant and can be backed up using an API call. See Overview of Block Volume for more information.

See Compute Shapes for information about which instance types support local NVMe storage.

Finding the NVMe devices on your instance

You can identify the NVMe devices by using the `lsblk` command. The response returns a list. NVMe devices begin with "`nvme`", as shown in the following example for a BM.DenseIO1.36 instance:

```
[opc@somehost ~]$ lsblk
NAME    MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda       8:0    0 46.6G  0 disk
├─sda1    8:1    0  512M  0 part /boot/efi
├─sda2    8:2    0    8G  0 part [SWAP]
└─sda3    8:3    0   38G  0 part /
nvme0n1 259:6    0  2.9T  0 disk
nvme1n1 259:8    0  2.9T  0 disk
nvme2n1 259:0    0  2.9T  0 disk
nvme3n1 259:1    0  2.9T  0 disk
nvme4n1 259:7    0  2.9T  0 disk
nvme5n1 259:4    0  2.9T  0 disk
nvme6n1 259:5    0  2.9T  0 disk
nvme7n1 259:2    0  2.9T  0 disk
nvme8n1 259:3    0  2.9T  0 disk
[opc@somehost ~]$
```

Failure Modes and How to Protect Against Them

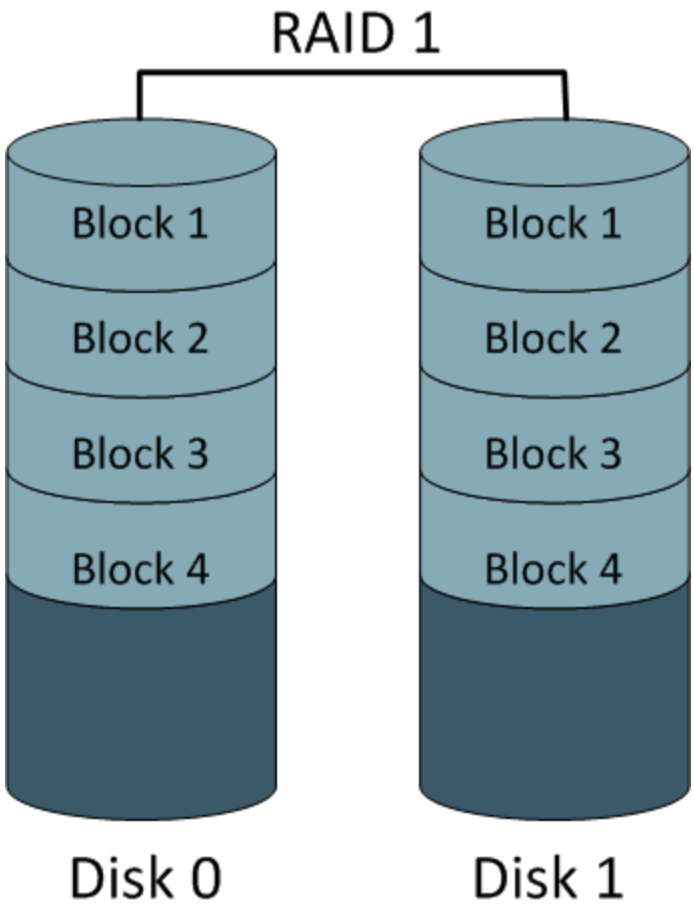
There are three primary failure modes you should plan for:

- Protecting against the failure of an NVMe device
- Protecting Against the Loss of the Instance or Availability Domain
- Protecting Against Data Corruption or Loss from Application or User Error

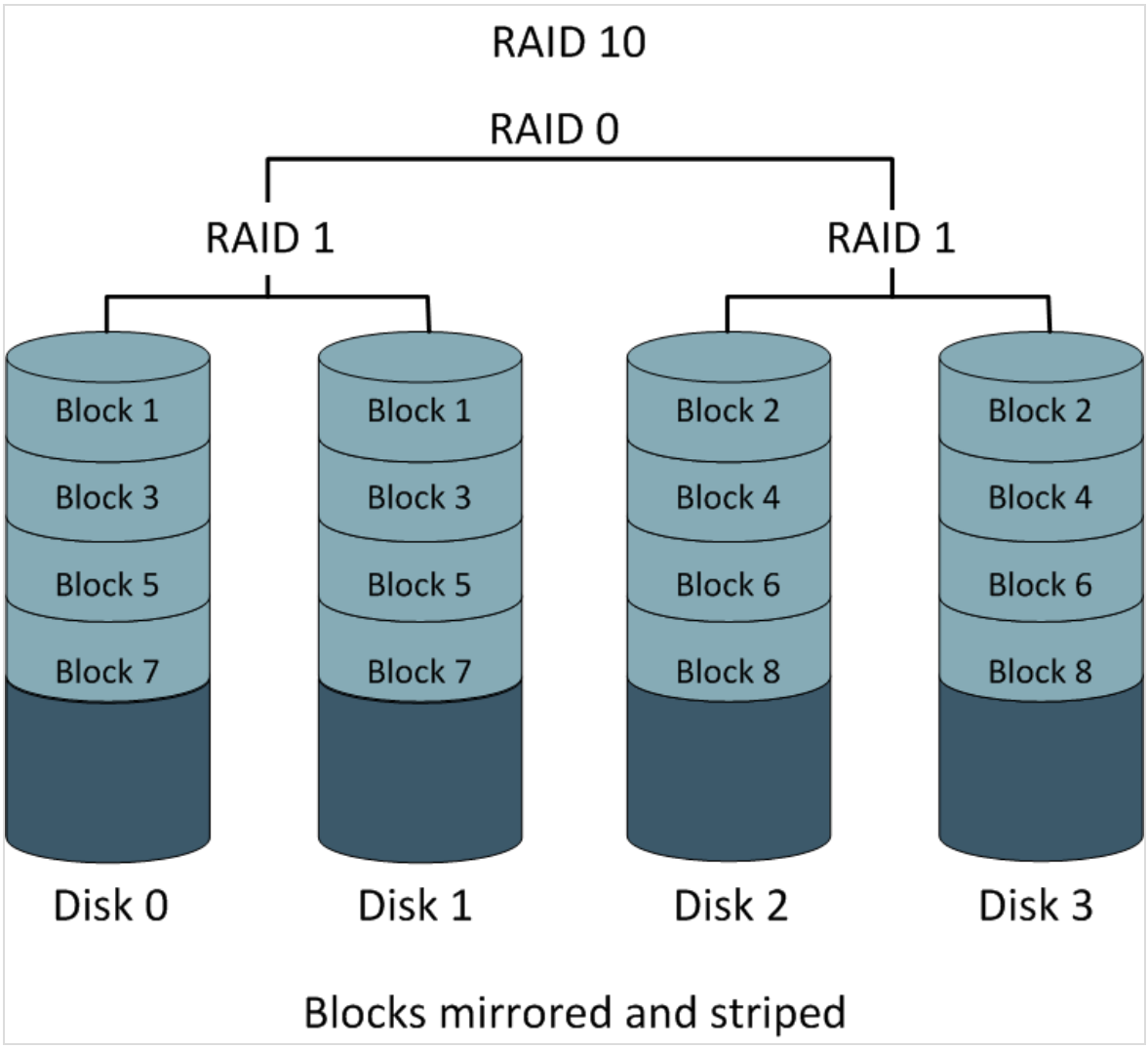
Protecting against the failure of an NVMe device

A protected RAID array is the most recommended way to protect against an NVMe device failure. There are three RAID levels that can be used for the majority of workloads:

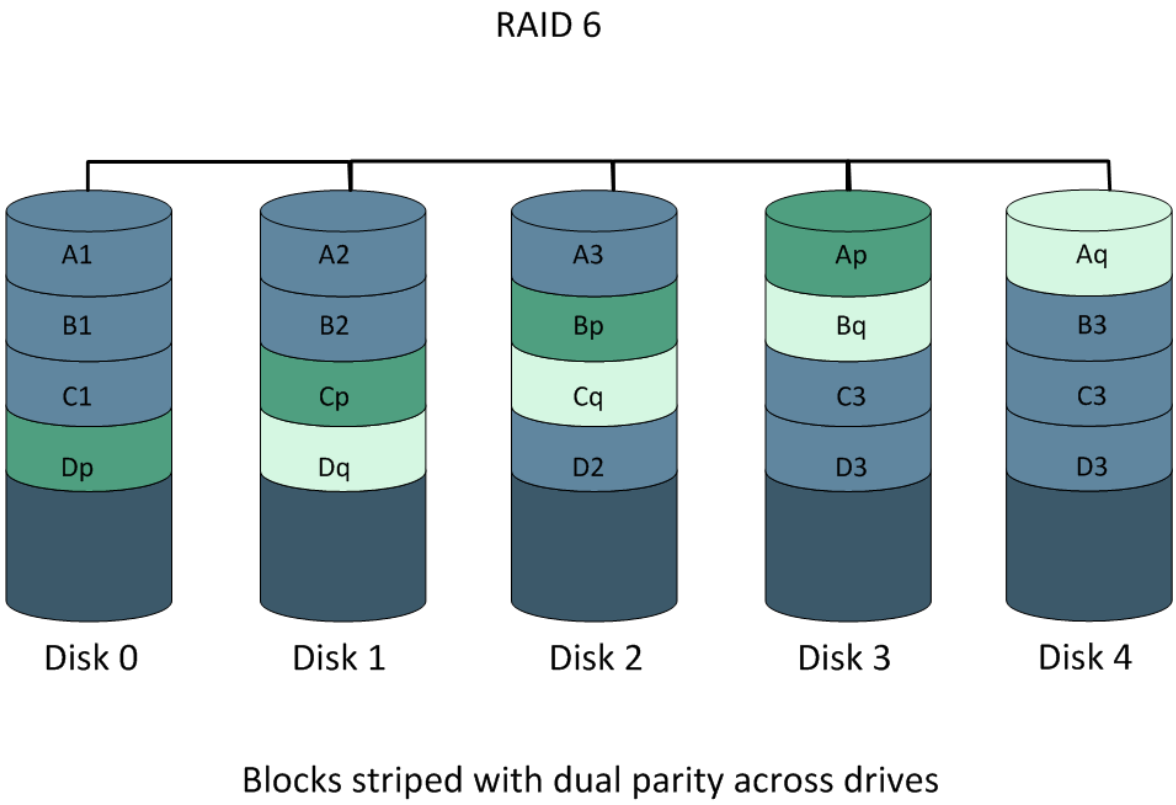
- RAID 1: An exact copy (or mirror) of a set of data on two or more disks; a classic RAID 1 mirrored pair contains two disks, as shown:



- RAID 10: Stripes data across multiple mirrored pairs. As long as one disk in each mirrored pair is functional, data can be retrieved, as shown:



- RAID 6: Block-level striping with two parity blocks distributed across all member disks, as shown.



For more information about RAID and RAID levels, see RAID.

Because the appropriate RAID level is a function of the number of available drives, the number of individual LUNs needed, the amount of space needed, and the performance requirements, there isn't one correct choice. You must understand your workload and design accordingly.

Options for Using a BM.DenseIO1.36 Shape

There are several options for BM.DenseIO1.36 instances with nine NVMe devices:

Create a single RAID 6 device across all nine devices. This array is redundant, performs well, will survive the failure of any two devices, and will be exposed as a single LUN with about 23.8TB of usable space.

Use the following commands to create a single RAID 6 device across all nine devices:

```
$ sudo yum install mdadm -y
```

```
$ sudo mdadm --create /dev/md0 --raid-devices=9 --level=6 /dev/nvme0n1 /dev/nvme1n1 /dev/nvme2n1
/dev/nvme3n1 /dev/nvme4n1 /dev/nvme5n1 /dev/nvme6n1 /dev/nvme7n1 /dev/nvme8n1
```

```
$ sudo mdadm --detail --scan | sudo tee -a /etc/mdadm.conf >> /dev/null
```

Create a four device RAID 10 and a five device RAID 6 array. These arrays would be exposed as two different LUNs to your applications. This is a recommended choice when you need to isolate one type of I/O from another, such as log and data files. In this example, your RAID 10 array would have about 6.4TB of usable space and the RAID 6 array would have about 9.6TB of usable space.

Use the following commands to create a four-device RAID 10 and a five-device RAID 6 array:

```
$ sudo yum install mdadm -y
```

```
$ sudo mdadm --create /dev/md0 --raid-devices=4 --level=10 /dev/nvme5n1 /dev/nvme6n1 /dev/nvme7n1
/dev/nvme8n1
```

```
$ sudo mdadm --create /dev/md1 --raid-devices=5 --level=6 /dev/nvme0n1 /dev/nvme1n1 /dev/nvme2n1
/dev/nvme3n1 /dev/nvme4n1
```

```
$ sudo mdadm --detail --scan | sudo tee -a /etc/mdadm.conf >> /dev/null
```

If you need the best possible performance and can sacrifice some of your available space, then an eight-device RAID 10 array is an option. Because RAID 10 requires an even number of devices, the ninth device is left out of the array and serves as a hot spare in case another device fails. This creates a single LUN with about 12.8 TB of usable space.

Use the following commands to create an eight-device RAID 10 array:

```
$ sudo yum install mdadm -y
```

```
$ sudo mdadm --create /dev/md0 --raid-devices=8 --level=10 /dev/nvme0n1 /dev/nvme1n1 /dev/nvme2n1
/dev/nvme3n1 /dev/nvme4n1 /dev/nvme5n1 /dev/nvme6n1 /dev/nvme7n1
```

The following command adds /dev/nvme8n as a hot spare for the /dev/md0 array:

```
$ sudo mdadm /dev/md0 --add /dev/nvme8n1
```

```
$ sudo mdadm --detail --scan | sudo tee -a /etc/mdadm.conf >> /dev/null
```

For the best possible performance and I/O isolation across LUNs, create two four-device RAID 10 arrays. Because RAID 10 requires an even number of devices, the ninth device is left out of the arrays and serves as a global hot spare in case another device in either array fails. This creates two LUNS, each with about 6.4 TB of usable space.

Use the following commands to create two four-device RAID 10 arrays with a global hot spare:

```
$ sudo yum install mdadm -y
```

```
$ sudo mdadm --create /dev/md0 --raid-devices=4 --level=10 /dev/nvme4n1 /dev/nvme5n1 /dev/nvme6n1
/dev/nvme7n1
```

```
$ sudo mdadm --create /dev/md1 --raid-devices=4 --level=10 /dev/nvme0n1 /dev/nvme1n1 /dev/nvme2n1
/dev/nvme3n1
```

Creating a global hot spare requires the following two steps:

1. Add the spare to either array (it does not matter which one) by running these commands:

   ```
   $ sudo mdadm /dev/md0 --add /dev/nvme8n1
   ```

   ```
   $ sudo mdadm --detail --scan | sudo tee -a /etc/mdadm.conf >> /dev/null
   ```

2. Edit `/etc/mdadm` to put both arrays in the same spare-group. Add `spare-group=global` to the end of the line that starts with `ARRAY`, as follows:

   ```
   $ sudo vi /etc/mdadm.conf
   ```

   ```
   ARRAY /dev/md0 metadata=1.2 spares=1 name=mdadm.localdomain:0
   UUID=43f93ce6:4a19d07b:51762f1b:250e2327 spare-group=global
   ```

   ```
   ARRAY /dev/md1 metadata=1.2 name=mdadm.localdomain:1 UUID=7521e51a:83999f00:99459a19:0c836693
   spare-group=global
   ```

Monitoring Your Array

It's important for you to be notified if a device in one of your arrays fails. Mdadm has built-in tools that can be utilized for monitoring, and there are two options you can use:

- Set the `MAILADDR` option in `/etc/mdadm.conf` and then run the `mdadm` monitor as a daemon

- Run an external script when `mdadm` detects a failure

SET THE MAILADDR OPTION IN /ETC/MDADM.CONF AND RUN THE MDADM MONITOR AS A DAEMON

The simplest method is to set the `MAILADDR` option in `/etc/mdadm.conf`, and then run the mdadm monitor as a daemon, as follows:

1. The `DEVICE partitions` line is required for `MAILADDR` to work; if it is missing, you must add it, as follows:

   ```
   $ sudo vi /etc/mdadm.conf

   DEVICE partitions

   ARRAY /dev/md0 level=raid1 UUID=1b70e34a:2930b5a6:016we78d:eese14532

   MAILADDR <my.name@oracle.com>
   ```

2. Run the monitor using the following command:

   ```
   $ sudo nohup mdadm --monitor --scan --daemonize &
   ```

3. To verify that the monitor runs at startup, run the following commands:

   ```
   $ sudo chmod +x /etc/rc.d/rc.local

   $ sudo vi /etc/rc.local
   ```

   Add the following line to the end of /etc/rc.local:

   ```
   nohup mdadm --monitor --scan --daemonize &
   ```

4. To verify that the email and monitor are both working run the following command:

   ```
   $ sudo mdadm --monitor --scan --test -1
   ```

   Note that these emails will likely be marked as spam. The `PROGRAM` option, described later in this topic, allows for more sophisticated alerting and messaging.

RUN AN EXTERNAL SCRIPT WHEN A FAILURE IS DETECTED

A more advanced option is to create an external script that would run if the `mdadm` monitor detects a failure. You would integrate this type of script with your existing monitoring solution. The following is an example of this type of script:

```
$ sudo vi /etc/mdadm.events
```

```
#!/bin/bash
event=$1
device=$2
if [ $event == "Fail" ]
then
  <"do something">
else
 if [ $event == "FailSpare" ]
 then
  <"do something else">
 else
  if [ $event == "DegradedArray" ]
  then
   <"do something else else">
  else
   if [ $event == "TestMessage" ]
    then
    <"do something else else else">
   fi
  fi
 fi
fi

$ sudo chmod +x /etc/mdadm.events
```

Next, add the `PROGRAM` option to `/etc/mdadm.conf`, as shown in the following example:

1. The `DEVICE partitions` line is required for `MAILADDR` to work; if it is missing, you must add it, as follows:

   ```
   $ sudo vi /etc/mdadm.conf
   ```

   ```
   DEVICE partitions
   ```

   ```
   ARRAY /dev/md0 level=raid1 UUID=1b70e34a:2930b5a6:016we78d:eese14532
   ```

   ```
   MAILADDR <my.name@oracle.com>
   ```

   ```
   PROGRAM /etc/mdadm.events
   ```

2. Run the monitor using the following command:

   ```
   $ sudo nohup mdadm --monitor --scan --daemonize &
   ```

3. To verify that the monitor runs at startup, run the following commands:

```
$ sudo chmod +x /etc/rc.d/rc.local
```

```
$ sudo vi /etc/rc.local
```

Add the following line to the end of /etc/rc.local:

```
nohup mdadm --monitor --scan --daemonize &
```

4. To verify that the email and monitor are both working run the following command:

```
$ sudo mdadm --monitor --scan --test -1
```

Note that these emails will likely be marked as spam. The `PROGRAM` option, described later in this topic, allows for more sophisticated alerting and messaging.

SIMULATE THE FAILURE OF A DEVICE

You can use `mdadm` to manually cause a failure of a device to see whether your RAID array can survive the failure, as well as test the alerts you have set up.

1. Mark a device in the array as failed by running the following command:

```
$ sudo mdadm /dev/md0 --fail /dev/nvme0n1
```

2. Recover the device or your array might not be protected. Use the following command:

```
$ sudo mdadm /dev/md0 --add /dev/nvme0n1
```

Your array will automatically rebuild in order to use the "new" device. Performance will be decreased during this process.

3. You can monitor the rebuild status by running the following command:

```
$ sudo mdadm --detail /dev/md0
```

What To Do When an NVMe Device Fails

Compute resources in the cloud are designed to be temporary and fungible. If an NVMe device fails while the instance is in service, you should start another instance with the same amount of storage or more, and then copy the data onto the new instance, replacing the old instance. There are multiple toolsets for copying large amounts of data, with rsync being the most popular. Since the connectivity between instances is a full 10Gb/sec, copying data should be quick. Remember that with a failed device, your array may no longer be protected, so you should copy the data off of the impacted instance as quickly as possible.

Using the Linux Logical Volume Manager

The Linux Logical Volume Manager (LVM) provides a rich set of features for managing volumes. If you need these features, we strongly recommend that you use `mdadm` as described in preceding sections of this topic to create the RAID arrays, and then use LVM's `pvcreate`, `vgcreate`, and `lvcreate` commands to create volumes on the `mdadm` LUNs. You should not use LVM directly against your NVMe devices.

Protecting Against the Loss of the Instance or Availability Domain

Once your data is protected against the loss of a NVMe device, you need to protect it against the loss of an instance or the loss of the availability domain. This type of protection is typically done by replicating your data to another availability domain or backing up your data to another location. The method you choose depends on your objectives. For details, see Recovery Time Objective (RTO) and Recovery Point Objective (RPO).

Replication

Replicating your data from one instance in one availability domain to another has the lowest RTO and RPO at a significantly higher cost than backups; for every instance in one availability domain, you must have another instance in a different availability domain.

For Oracle database workloads, you should use the built-in Oracle Data Guard functionality to replicate your databases. Oracle Cloud Infrastructure availability domains are each close enough to each other to support high performance, synchronous replication. Asynchronous replication is also an option.

For general-purpose block replication, DRBD is the recommended option. You can configure DRBD to replicate, synchronously or asynchronously, every write in one availability domain to another availability domain.

Backups

Traditional backups are another way to protect data. All commercial backup products are fully supported on Oracle Cloud Infrastructure. If you use backups, the RTO and RPO are significantly higher than using replication because you must recreate the compute resources that failed and then restore the most recent backup. Costs are significantly lower because you

don't need to maintain a second instance. Do not store your backups in the same availability domain as their original instance.

Protecting Against Data Corruption or Loss from Application or User Error

The two recommended ways of protecting against data corruption or loss from application or user error are regularly taking snapshots or creating backups.

SNAPSHOTS

The two easiest ways to maintain snapshots are to either use a file system that supports snapshots, such as ZFS, or use LVM to create and manage the snapshots. Because of the way LVM has implemented copy-on-write (COW), performance may significantly decrease when a snapshot is taken using LVM.

BACKUPS

All commercial backup products are fully supported on Oracle Cloud Infrastructure. Make sure that your backups are not stored in the same availability domain as their original instance.

Boot Volumes

When you launch a virtual machine (VM) or bare metal instance based on an Oracle-provided image or custom image, a new boot volume for the instance is created in the same compartment. That boot volume is associated with that instance until you terminate the instance. When you terminate the instance, you can preserve the boot volume and its data. For more information, see Terminating an Instance. This feature gives you more control and management options for your compute instance boot volumes, and enables:

- **Instance scaling:** When you terminate your instance, you can keep the associated boot volume and use it to launch a new instance using a different instance type or shape. See Creating an Instance for steps to launch an instance based on a boot volume. This allows you to switch easily from a bare metal instance to a VM instance and vice versa, or scale up or down the number of cores for an instance.
- **Troubleshooting and repair:** If you think a boot volume issue is causing a compute instance problem, you can stop the instance and detach the boot volume. Then you can

attach it to another instance as a data volume to troubleshoot it. After resolving the issue, you can then reattach it to the original instance or use it to launch a new instance.

Boot volumes are encrypted by default, the same as other block storage volumes. For more information, see Block Volume Encryption.



Important

In-transit encryption for boot and block volumes is only available for virtual machine (VM) instances launched from Oracle-provided images, it is not supported on bare metal instances. It is also not supported in most cases for instances launched from custom images imported for "bring your own image" (BYOI) scenarios. To confirm support for certain Linux-based custom images and for more information contact Oracle support, see Contacting Support.

You can group boot volumes with block volumes into the same volume group, making it easy to create a group volume backup or a clone of your entire instance, including both the system disk and storage disks at the same time. See Volume Groups for more information.

For more information about the Block Volume service and boot volumes, see the Block Volume FAQ.

Custom Boot Volume Sizes

When you launch an instance, you can specify whether to use the selected image's default boot volume size, or you can specify a custom size up to 32 TB. This capability is available for the following image source options:

- Oracle-provided image
- Custom image
- Image OCID

See Creating an Instance for more information.

The specified size must be larger than the image's default boot volume size or 50 GB, whichever is higher. After you launch the instance, you can't change the boot volume size.

If you specify a custom boot volume size, you need to extend the volume to take advantage of the larger size. For steps, see Extending a Root or System Partition.

Required IAM Service Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let users launch Compute instances includes the ability to list boot volumes. The policy in Let volume admins manage block volumes, backups, and volume groups lets the specified group do everything with block volumes, boot volumes, and backups, but not launch instances.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Using the Console

To access the Console, you must use a supported browser.

See the following tasks for managing boot volumes:

- Listing Boot Volumes
- Attaching a Boot Volume
- Detaching a Boot Volume

- Listing Boot Volume Attachments
- Deleting a Boot Volume

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage boot volumes:

- BootVolume
- ListBootVolumes
- GetBootVolume
- UpdateBootVolume
- DetachBootVolume
- DeleteVolume
- BootVolumeAttachment
- AttachBootVolume
- GetBootVolumeAttachment
- ListBootVolumeAttachments

Extending the Partition for a Boot Volume

When you create a new virtual machine (VM) instance or bare metal instance based on an Oracle-provided image or custom image, you have the option of specifying a custom boot volume size. You can also expand the size of the boot volume for an existing instance; see Resizing a Volume for more information. In order to take advantage of the larger size, you need to extend the partition for the boot volume.

Required IAM Policy

Extending a partition on an instance does not require a specific IAM policy. However, you may need permission to run the necessary commands on the instance's guest OS. Contact your system administrator for more information.

Extending the Root Partition on a Linux-Based Image

For instances running Linux-based images, you need to extend the root partition and then grow the file system.

PREREQUISITES

Before you can extend the partition, you need to detach the boot volume from the instance and attach it to another instance as a data volume. To do this:

1. Stop the instance. For steps, see Stopping and Starting an Instance.
2. After the instance has stopped, detach the boot volume. For steps, see Detaching a Volume.
3. Attach the boot volume to a second instance as a data volume. For steps, see Attaching a Volume and Connecting to a Volume.

EXTENDING THE LINUX PARTITION

After attaching the boot volume as a data volume to the second instance, connect to this instance and perform the following steps to extend the partition.

Extending the root partition

1. To identify the volume you want to extend the partition for, run the following command to list the attached block volumes:

```
lsblk
```

2. Run the following command to edit the volume's partition table with `parted`:

```
parted <volume_id>
```

   <volume_id> is the volume identifier, for example `/dev/sdc`.

3. When you run `parted`, you may encounter the following error message:

```
Warning: Not all of the space available to <volume_id> appears to be used,
you can fix the GPT to use all of the space (an extra volume_size blocks)
or continue with the current setting?
```

   You are then prompted to fix the error or ignore the error and continue with the current setting. Specify the option to fix the error.

4. Run the following command to change the display units to sectors so that you can see the precise start position for the volume:

```
(parted) unit s
```

5. Run the following command to display the current partitions in the partition table:

```
(parted) print
```

   Make note of the values in the **Number**, **Start**, and **File system** columns for the root partition.

6. Run the following command to remove the existing root partition:

```
(parted) rm <partition_number>
```

   <partition_number> is the value from the **Number** column.

7. Run the following command to recreate the partition:

```
(parted) mkpart
```

   At the `Start?` prompt, specify the value from the **Start** column. At the `File system type?` prompt, specify the value from the **File system** column. Specify `100%` for the `End?` prompt.

8. Run the following command to exit `parted`:

   ```
   (parted) quit
   ```

   This command forces a rewrite of the partition table with the new partition settings that you specified.

9. To verify that the root partition was extended, run the following command to list the attached block volumes:

   ```
   lsblk
   ```

After you extend the root partition, you need to grow the file system. The steps in the following procedure apply only to xfs file systems.

Growing the file system for the root partition

1. Before you grow the file system, repair any issues with the file system on the extended partition by running the following command:

   ```
   xfs_repair <partition_id>
   ```

   <partition_id> is the partition identifier, for example `/dev/sdc3`. See Checking and Repairing an XFS File System for more information.

2. After you have confirmed that there are no more issues to repair, you need to create a mount point to run the `xfs_growfs` against. To do this, create a directory and mount the partition to that directory by running the following commands:

   ```
   mkdir <directory_name>
   mount <partition_id> <directory_name> -o nouuid
   ```

   <partition_id> is the partition identifier, for example `/dev/sdc3`, and *<directory_name>* is the directory to create and mount.

3. After you have created the mount point run the following command to grow the file system:

   ```
   xfs_growfs -d <directory_name>
   ```

   <directory_name> is the name for the directory you created in the previous step.

4. To verify that the file system size is correct, run the following command to display the file system details:

   ```
   df -lh
   ```

5. Once you have verified that the file system size is correct, run the following command to unmount the partition:

   ```
   umount <partition_id>
   ```

Next Steps

After you have extended the partition and grown the file system, you can restart the original instance with the boot volume. To do this:

1. Disconnect the volume from the second instance. For steps, see Disconnecting From a Volume.
2. Detach the volume from the second instance. For steps, see Detaching a Volume.
3. Attach the volume to the original instance as a boot volume. For steps, see Attaching a Boot Volume.
4. Restart the instance. For steps, see Stopping and Starting an Instance.

Extending the System Partition on a Windows-Based Image

On Windows-based images, you can extend a partition using the Windows interface or from the command line using the DISKPART utility.

Windows Server 2016 and Windows Server 2012

The steps for extending a system partition on instances running Windows 2012 or Windows 2016 are the same, and are described in the following procedures.

Extending the system partition using the Windows interface

1. Open the Disk Management system utility on the instance.
2. Right-click the boot volume and select **Extend Volume**.
3. Follow the instructions in the **Extend Volume Wizard**:
   a. Select the disk that you want to extend, enter the size, and then click **Next**.
   b. Confirm that the disk and size settings are correct, and then click **Finish**.
4. Verify that the boot volume's system disk has been extended in Disk Management.

Extending the system partition using the command line with DISKPART

1. Open a command prompt as administrator on the instance.
2. Run the following command to start the DISKPART utility:

   ```
   diskpart
   ```

3. At the `DISKPART` prompt, run the following command to display the instance's volumes:

   ```
   list volume
   ```

4. Run the following command to select the boot volume:

   ```
   select volume <volume_number>
   ```

   <volume_number> is the number associated with the boot volume that you want to extend the partition for.

5. Run the following command to extend the partition:

```
extend size=<increased_size_in_MB>
```

<increased_size_in_MB> is the size in MB that you want to extend the partition to.



Warning

When using the DISKPART utility, do not overextend the partition beyond the current available space. Overextending the partition could result in data loss.

6. To confirm that the partition was extended, run the following command and verify that the boot volume's partition has been extended:

```
list volume
```

WINDOWS SERVER 2008

Use the steps described in the following procedures to extend a system partition on instances running Windows 2008.

Extending the system partition using the Windows interface

1. Open the Server Manager on the instance.
2. Expand the **Storage** node in the left navigation pane and click **Disk Management**.
3. Right-click the boot volume and select **Extend Volume**.
4. Follow the instructions in the **Extend Volume Wizard**:
    a. Select the disk that you want to extend, enter the size, and then click **Next**.
    b. Confirm that the disk and size settings are correct, and then click **Finish**.
5. Verify that the boot volume's system disk has been extended in the Server Manager's **Disk Management** node.

Extending the system partition using the command line with DISKPART

1. Open a command prompt as administrator on the instance.
2. Run the following command to start the DISKPART utility:

```
diskpart
```

3. At the `DISKPART` prompt, run the following command to display the instance's volumes:

```
list volume
```

4. Run the following command to select the boot volume:

```
select volume <volume_number>
```

<volume_number> is the number associated with the boot volume that you want to extend the partition for.

5. Run the following command to extend the partition:

```
extend size=<increased_size_in_MB>
```

<increased_size_in_MB> is the size in MB that you want to extend the partition to.



Warning

When using the DISKPART utility, do not overextend the partition beyond the current available space. Overextending the partition could result in data loss.

6. To confirm that the partition was extended, run the following command and verify that the boot volume's partition has been extended:

```
list volume
```

Attaching a Boot Volume

If a boot volume has been detached from the associated instance, or if the instance is stopped or terminated, you can attach it to another instance as a data volume. The steps are the same as the steps for attaching a block volume, see Attaching a Volume.

You can also reattach a boot volume to the associated instance. If you want to restart an instance with a detached boot volume, you must reattach the boot volume using the steps described in this topic.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let users launch Compute instances includes the ability to attach/detach existing block volumes. The policy in Let volume admins manage block volumes, backups, and volume groups lets the specified group do everything with block volumes and backups, but not launch instances.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
2. In the **Instances** list, select the instance you want to attach the boot volume to.
3. Click the name of the instance to display the instance details.
4. In the **Resources**, click **Boot Volume**.
5. Click the Actions icon (three dots) for the boot volume.

6. Click **Attach** and confirm the selection when prompted.
   You can start the instance once the boot volume's icon no longer lists it as **ATTACHING**. For more information, see Stopping and Starting an Instance.

Using the API

To attach a volume to an instance, use the following operation:

- AttachBootVolume

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Listing Boot Volumes

You can list all boot volumes in a specific compartment, or detailed information on a single boot volume.

Required IAM Service Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let users launch Compute instances includes the ability to list volumes. The policy in Let volume admins manage block volumes, backups, and volume groups lets the specified group do everything with block volumes and backups, but not launch instances.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Boot Volumes**.
2. Choose your **Compartment**.

A detailed list of volumes in the current compartment is displayed. To see detailed information for a specific volume, click the boot volume name.

The instance associated with the boot volume is listed in the **Attached Instance** field. If the value for this field displays:

```
None in this Compartment.
```

the boot volume has been detached from the associated instance, or the instance has been terminated while the boot volume was preserved.

To view the volumes in a different compartment, change the compartment in the **Compartment** drop-down menu.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

List Boot Volumes:

Get a list of boot volumes within a compartment.

- ListBootVolumes

Get a Single Boot Volume:

Get detailed information on a single boot volume:

- GetBootVolume

Overview of Boot Volume Backups

The backups feature of the Oracle Cloud Infrastructure Block Volume service lets you make a point-in-time crash-consistent backup of a boot volume without application interruption or downtime. Boot volume backup capabilities are the same as block volume backup capabilities. See Overview of Block Volume Backups for more information.

There are two ways you can initiate a boot volume backup, the same as block volume backups. You can either manually start the backup, or assign a policy which defines a set backup schedule. See Manual Backups and Policy-Based Backups for more information.

Boot Volume Backup Types

The Block Volume service supports the same backups types for boot volumes as for block volumes:

- **Incremental:** This backup type includes only the changes since the last backup.
- **Full:** This backup type includes all changes since the volume was created.

Backing Up a Boot Volume

You can create boot volume backups using the Console or the REST APIs/command line interface (CLI). See Backing Up a Boot Volume and the BootVolumeBackup API for more information.

Restoring a Boot Volume

Before you can use a boot volume backup, you need to restore it. For steps, see Restoring a Boot Volume.

Making a boot volume backup while an instance is running creates a crash-consistent backup, meaning the data is in the identical state it was in at the time the backup was made. This is the same state it would be in the case of a loss of power or hard crash. In most cases, you can restore a boot volume backup and use it to create an instance. Alternatively you can attach it to an instance as a data volume to repair it or recover data, see Attaching a Volume. To

ensure a bootable image, you should create a custom image from your instance. For information about creating custom images, see Managing Custom Images.

Differences Between Boot Volume Backups and Clones

Consider the following criteria when you decide whether to create a backup or a clone of a volume.

	Volume Backup	Volume Clone
Description	Creates a point-in-time backup of data on a volume. You can restore multiple new volumes from the backup later in the future.	Creates a single point-in-time copy of a volume without having to go through the backup and restore process.
Use case	Retain a backup of the data in a volume, so that you can duplicate an environment later or preserve the data for future use. Meet compliance and regulatory requirements, because the data in a backup remains unchanged over time. Support business continuity requirements. Reduce the risk of outages or data mutation over time.	Rapidly duplicate an existing environment. For example, you can use a clone to test configuration changes without impacting your production environment.
Speed	Slower (minutes or hours)	Faster (seconds)
Cost	Lower cost	Higher cost
Storage location	Object Storage	Block Volume

	Volume Backup	Volume Clone
Retention policy	Policy-based backups expire, manual backups do not expire	No expiration
Volume groups	Supported. You can back up a volume group.	Supported. You can clone a volume group.

Backing Up a Boot Volume

You can create a backup of a boot volume using the Oracle Cloud Infrastructure Block Volume service. This topic describes how to create a manual boot volume backup.

You can also configure a backup policy that creates backups automatically based on a specified schedule and retention policy. This works the same as block volumes. See Policy-Based Backups for more information.

For information to help you decide whether to create a backup or a clone of a boot volume, see Differences Between Boot Volume Backups and Clones.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.



Tip

When users create a backup from a volume or restore a volume from a backup, the volume and backup don't have to be in the same compartment. However, users must have access to both compartments.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Boot Volumes**.
2. Click the boot volume that you want to create a backup for.
3. Click **Create Manual Backup**.
4. Enter a name for the backup.
5. Select the backup type, either incremental or full. See Boot Volume Backup Types for information about backup types.
6. Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
7. Click **Create Backup**.
   The backup is completed when its icon no longer lists it as **CREATING** in the **Boot Volume Backup** list.

Using the API

To back up a boot volume, use the following operation:

- CreateBootVolumeBackup

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

For more information about backups, see Overview of Block Volume Backups and Restoring a Backup to a New Volume.

Cloning a Boot Volume

You can create a clone from a boot volume using the Oracle Cloud Infrastructure Block Volume service. Cloning enables you to make a copy of an existing boot volume without needing to go through the backup and restore process. For more information about the Block Volume service, see Overview of Block Volume and the Block Volume FAQ.

A boot volume clone is a point-in-time direct disk-to-disk deep copy of the source boot volume, so all the data that is in the source boot volume when the clone is created is copied to the boot volume clone. Any subsequent changes to the data on the source boot volume are not copied to the boot volume clone. Since the clone is a copy of the source boot volume it will be the same size as the source boot volume unless you specify a larger volume size when you create the clone.

The clone operation occurs immediately and you can use the cloned boot volume as soon as the state changes to available.

There is a single point-in-time reference for a source boot volume while it is being cloned, so if you clone a boot volume while the associated instance is running, you need to wait for the first clone operation to complete from the source before creating additional clones. You also need to wait for any backup operations to complete as well.

You can only create a clone for a boot volume within the same region, availability domain, and tenant. You can create a clone for a boot volume between compartments as long as you have the required access permissions for the operation.

Differences Between Boot Volume Backups and Clones

Consider the following criteria when you decide whether to create a backup or a clone of a volume.

	Volume Backup	Volume Clone
Description	Creates a point-in-time backup of data on a volume. You can restore multiple new volumes from the backup later in the future.	Creates a single point-in-time copy of a volume without having to go through the backup and restore process.
Use case	Retain a backup of the data in a volume, so that you can duplicate an environment later or preserve the data for future use. Meet compliance and regulatory requirements, because the data in a backup remains unchanged over time. Support business continuity requirements. Reduce the risk of outages or data mutation over time.	Rapidly duplicate an existing environment. For example, you can use a clone to test configuration changes without impacting your production environment.
Speed	Slower (minutes or hours)	Faster (seconds)
Cost	Lower cost	Higher cost
Storage location	Object Storage	Block Volume

	Volume Backup	Volume Clone
Retention policy	Policy-based backups expire, manual backups do not expire	No expiration
Volume groups	Supported. You can back up a volume group.	Supported. You can clone a volume group.

For more information about boot volume backups, see Overview of Boot Volume Backups and Backing Up a Boot Volume.

Using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Boot Volumes**.
2. In the **Boot Volumes** list, click the boot volume that you want to clone.
3. In **Resources**, click **Boot Volume Clones**.
4. Click **Create Clone**.
5. Specify a name for the clone.
6. If you want to clone the boot volume to a larger size volume, select **Custom Boot Volume Size (GB)** and then specify the new size. You can only increase the size of the volume, you cannot decrease the size. If you clone the boot volume to a larger size volume, you need to extend the volume's partition. See Extending the Partition for a Boot Volume for more information.
7. Click **Create Clone**.

The boot volume is ready use when its icon lists it as **AVAILABLE** in the **Boot Volumes** list.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

To create a clone from a boot volume, use the CreateBootVolume operation and specify BootVolumeSourceFromBootVolumeDetails for CreateBootVolumeDetails.

Next Steps

After you have cloned a boot volume backup, you can:

- Use the boot volume to create an instance. For more information, see Creating an Instance.
- Attach the boot volume to an instance as a data volume. For more information, see Attaching a Volume.

Making a boot volume clone while an instance is running creates a crash-consistent clone, meaning the data is in the identical state it was in at the time the clone was made. This is the same state it would be in the case of a loss of power or hard crash. In most cases you can use the cloned boot volume to create an instance, however to ensure a bootable image, you should create a custom image from your instance. For information about creating custom images, see Managing Custom Images.

Detaching a Boot Volume

If you think a boot volume issue is causing a compute instance problem, you can stop the instance and detach the boot volume using the steps described in this topic. Then you can attach it to another instance as a data volume to troubleshoot it.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let users launch Compute instances includes the ability to attach/detach existing block volumes. The policy in Let volume admins manage block

volumes, backups, and volume groups lets the specified group do everything with block volumes and backups, but not launch instances.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Using the Console

You can only detach a boot volume from an instance when the instance is stopped. See Stopping and Starting an Instance for more information about managing an instance's state.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
2. Choose your **Compartment**.
3. In the **Instances** list, select the instance you want to detach the boot volume from.
4. Click the name of the instance to display the instance details.
5. In the **Resources**, click **Boot Volume**.
6. Click the Actions icon (three dots), for the boot volume.
7. Click **Detach** and confirm the selection when prompted.

You can now attach the boot volume to another instance, for more information see Attaching a Volume.

Using the API

To delete an attachment, use the following operation:

- DetachBootVolume

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Deleting a Boot Volume

When you terminate an instance, you choose to delete or preserve the associated boot volume. For more information, see Terminating an Instance. You can also delete a boot volume if it has been detached from the associated instance. See Detaching a Boot Volume for how to detach a boot volume.



Warning

You cannot undo this operation. Any data on a volume will be permanently deleted once the volume is deleted. You will also not be able to restart the associated instance.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let volume admins manage block volumes, backups, and volume groups lets the specified group do everything with block volumes and backups.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Boot Volumes**.
2. Choose your **Compartment**.

3. In the **Boot Volumes** list, find the volume you want to delete.
4. Click the Actions icon (three dots) for the boot volume.
5. Click **Terminate** and confirm the selection when prompted.

Using the API

Use the DeleteBootVolume operation to delete a boot volume.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Boot Volume Metrics

You can monitor the health, capacity, and performance of your Compute instances by using metrics, alarms, and notifications.

The Block Volume service provides a set of metrics that apply to both boot volumes and block volumes. For more information, see Block Volume Metrics.

Recovering a Corrupted Boot Volume for Linux-Based Instances

If your instance fails to boot successfully or boots with the boot volume set to read-only access, the instance's boot volume may be corrupted. While it is rare, boot volume corruption can occur in the following scenarios:

- When an instance experiences a forced shutdown using the API.
- When an instance experiences a system hang due to an operating system or software error and a graceful reboot or shutdown of the instance times out, and then a forced shutdown occurs.
- When an error or outage occurs in the underlying infrastructure and there were critical disk writes pending in the system.



Important

In most cases a simple reboot will resolve boot volume corruption issues, so this is the first action you should take when troubleshooting this.

This topic describes how to determine if your Linux-based instance's boot volume is corrupted and what steps to take to troubleshoot and recover the corrupted boot volume. For Windows instances, see Recovering a Corrupted Boot Volume for Windows Instances.

Detecting Boot Volume Corruption

Boot volume corruption can prevent an instance from booting successfully, so you may not be able to connect to the instance using SSH. Instead, you can use the instance console connection feature to connect to the malfunctioning instance. For more information about using this feature, see Instance Console Connections.

This section describes how to use a serial console connection to detect if boot volume corruption has occurred.



Tip

If you have already confirmed your instance's boot volume is corrupted or if you are using an imported custom image, proceed to the Recovering the Boot Volume section, which describes how to use a second instance along with standard file system tools to both detect and repair boot volume corruption.

1. Create a serial console connection for the instance.

   ### To create the serial console connection for an instance

   a. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.

   b. In the list of instances, find the instance you think may have a corrupted boot volume, and then click the instance name.

   c. In the **Resources** section on the **Instance Details** page, click **Console Connections**, and then click **Create Console Connection**.

   d. Specify the public key portion for the SSH key, either by browsing and selecting a public key file, for example id_rsa.pub, or by pasting your public key into the text box, and then click **Create Console Connection**.

2. Connect to the instance through serial console.

   ### To connect to the serial console for an instance using OpenSSH on Mac OS X or Linux

   a. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**. Click the instance you want to connect to.

   b. On the **Instances Details** page, in the **Resources** section, click **Console Connections**.

   c. Click the Actions icon (three dots), and then click **Connect with SSH**.

   d. Select **LINUX/MAC OS** for **PLATFORM**.

   e. Click **Copy** to copy the string to the clipboard.

   f. Paste the connection string copied from the previous step to a terminal window on a Mac OS X or Linux system, and hit enter to connect to the console.

      If you are not using the default SSH key or ssh-agent, you can modify the serial console connection string to include the identity file flag, `-i` to specify the SSH

key to use. You need to specify this for both the SSH connection and the SSH ProxyCommand, as shown in the following line:

```
ssh -i /<path>/<ssh_key> -o ProxyCommand='ssh -i /<path>/<ssh_key> -W %h:%p -p 443...
```

g. Hit enter again to activate the console.

At this point, it's normal for the serial console to appear to hang, as the system may have already crashed.

3. Reboot the instance from the Console, on the **Instances Details page**, click **Reboot**.
4. Once the reboot process starts, switch back to the terminal window, and you should see system messages from the instance start to appear in the window.
5. Monitor the messages that appear as the system is starting up. Most operating systems will set the boot volume to read-only as soon as disk corruption is detected to prevent writes from further corrupting the volume, so look for messages that indicate the boot volume is in read-only mode. Following are some examples:
   - On an instance with iSCSI-attached boot volumes, the `iscsiadm` service will fail to attach a volume because the volume is in read-only mode. This will typically prevent instances from continuing to boot. The serial console may display a message similar to the following:

     ```
     iscsiadm: Maybe you are not root?
     iscsiadm: Could not lock discovery DB: /var/lock/iscsi/lock.write: Read-only file system
     touch: cannot touch `/var/lock/subsys/iscsid': Read-only file system
     touch: cannot touch `/var/lock/subsys/iscsi': Read-only file system
     ```

   - On an instance with paravirtualized-attached boot volumes, the system may continue the boot process, but will be in a degraded state because nothing can be written to the boot drive.The serial console may display error messages similar to the following:

     ```
     [FAILED] Failed to start Create Volatile Files and Directories.
     See 'systemctl status systemd-tmpfiles-setup.service' for details.
     ...
     [   27.160070] cloud-init[819]:     os.chmod(path, real_mode)
     [   27.166027] cloud-init[819]: OSError: [Errno 30] Read-only file system:
     '/var/lib/cloud/data'
     ```

The error messages and system behavior described here are the most commonly seen for boot volume corruption, however depending on the operating system, you may see different error messages and system behavior. If you don't see the ones described here, consult the documentation for your operating system for additional troubleshooting information.

Recovering the Boot Volume

To troubleshoot and recover the corrupted boot volume, you need to detach the boot volume from the instance and then attach the boot volume to a second instance as a data volume.

DETACHING THE BOOT VOLUME

If you have detected that your instance's boot volume is corrupted, you need to detach the boot volume from the instance before you can begin troubleshooting and recovery steps.

1. Stop the instance. For more information, see Stopping and Starting an Instance.

   #### To stop the instance

   a. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.

   b. In the list of instances, find the instance you have detected boot volume corruption for and then click the instance name to display the instance details.

   c. Click **Stop**.

2. Detach the boot volume from the instance. For more information, see Detaching a Boot Volume.

   #### To detach the boot volume from the instance

   a. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.

   b. In the **Instances** list, select the instance you want to detach the boot volume from.

c. Click the name of the instance to display the instance details.

d. In the **Resources**, click **Boot Volume**.

e. Click the Actions icon (three dots), for the boot volume.

f. Click **Detach** and confirm the selection when prompted.

ATTACHING THE BOOT VOLUME AS A DATA VOLUME TO A SECOND INSTANCE

For the second instance we recommend that you use an instance running an operating system that most closely matches the operating system for the boot volume's instance. You should only attach boot volumes for Linux-based instances to other Linux-based instances. The second instance must be in the same availability domain and region as the boot volume's instance. If no existing instance is available, create a new Linux instance using the steps described in Creating an Instance. Once you have the second instance, make sure you can log into the instance and that it is functional before proceeding with the recovery steps. For steps to access the instance, see Connecting to a Linux Instance. After you have confirmed that the instance is functional, perform the following steps.

1. Run the `lsblk` command and make note of the drives that are currently on the instance, for example:

```
lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda      8:0    0 46.6G  0 disk
├─sda2   8:2    0    8G  0 part [SWAP]
├─sda3   8:3    0 38.4G  0 part /
└─sda1   8:1    0  200M  0 part /boot/efi
```

2. Attach the boot volume to the second instance as a data volume. For more information, see Attaching a Volume.

To attach the boot volume as a data volume

a. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.

b. In the **Instances** list, click the instance that you want to attach a volume to.

c. In the **Resources** section, click **Attached Block Volumes**.

d. Click **Attach Block Volume**.

e. Select the volume attachment type. If **Paravirtualized** attachments are available for this instance, we recommend that you select this attachment type for this procedure.

If you select **iSCSI** as the volume attachment type, you need to connect to the volume, see Connecting to a Volume for more information.

f. In the **Block Volume Compartment** drop-down list, select the compartment.

g. Choose the **SELECT VOLUME** option and then select the volume from the **Boot Volume** section of the **Block Volume** drop-down list.

h. Select **Read/Write** as the access type.

i. Click **Attach**.

When the volume's icon no longer lists it as **Attaching**, proceed with the next steps.

3. Run the `lsblk` command again to confirm that the boot volume now shows up as a volume attached to the instance. In this sample output for the `lsblk`, the boot volume attached as a data volume shows up as `sdb`:

```
lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sdb      8:16   0 46.6G  0 disk
├─sdb2   8:18   0    8G  0 part
├─sdb3   8:19   0 38.4G  0 part
└─sdb1   8:17   0  200M  0 part
sda      8:0    0 46.6G  0 disk
├─sda2   8:2    0    8G  0 part [SWAP]
├─sda3   8:3    0 38.4G  0 part /
└─sda1   8:1    0  200M  0 part /boot/efi
```

4. Run the `fsck` command on the volume's root partition. The root partition is usually the largest partition on the volume.

The following sample for the `fsck` command shows the output when there are no errors or corruption present on the partitions for an Oracle 7.6 instance:

```
sudo fsck -V /dev/sdb1
fsck from util-linux 2.23.2
[/sbin/fsck.vfat (1) -- /boot/efi] fsck.vfat /dev/sdb1
fsck.fat 3.0.20 (12 Jun 2013)
/dev/sdb1: 17 files, 2466/51145 clusters

sudo fsck -V /dev/sdb2
fsck from util-linux 2.23.2

sudo fsck -V /dev/sdb3
fsck from util-linux 2.23.2
[/sbin/fsck.xfs (1) -- /] fsck.xfs /dev/sdb3
If you wish to check the consistency of an XFS filesystem or
repair a damaged filesystem, see xfs_repair(8).
```

If errors are present on a partition, you will usually be prompted to repair the errors. Following is an example of an interactive repair session of a corrupt ext4 boot volume for an Ubuntu instance:

```
sudo fsck -V /dev/sdb1
fsck from util-linux 2.31.1
[/sbin/fsck.ext4 (1) -- /] fsck.ext4 /dev/sdb1
e2fsck 1.44.1 (24-Mar-2018)
One or more block group descriptor checksums are invalid.  Fix<y> yes
Group descriptor 92 checksum is 0xe9a1, should be 0x1f53.  FIXED.
Pass 1: Checking inodes, blocks, and sizes
Pass 2: Checking directory structure

Pass 3: Checking directory connectivity
Pass 4: Checking reference counts
Pass 5: Checking group summary information
Block bitmap differences: Group 92 block bitmap does not match checksum.
FIXED.

cloudimg-rootfs: ***** FILE SYSTEM WAS MODIFIED *****
cloudimg-rootfs: 75336/5999616 files (0.1% non-contiguous), 798678/12181243 blocks
```



Note

XFS file systems will usually auto-repair their contents when the system boots up, fixing any corruption during the boot process. You can use the `xfs_repair` command to force a repair for scenarios where boot volume corruption is preventing the auto-repair functionality from working, as shown in the following example:

```
sudo xfs_repair /dev/sdb3
Phase 1 - find and verify superblock...
Phase 2 - using internal log
- zero log...
- scan filesystem freespace and inode maps...
...
Phase 7 - verify and correct link counts...
done
```

Recovering a Corrupted Boot Volume for Windows Instances

If your instance fails to boot successfully or boots with the boot volume set to read-only access, the instance's boot volume may be corrupted. While it is rare, boot volume corruption can occur in the following scenarios:

- When an instance experiences a forced shutdown using the API.
- When an instance experiences a system hang due to an operating system or software error and a graceful reboot or shutdown of the instance times out, and then a forced shutdown occurs.
- When an error or outage occurs in the underlying infrastructure and there were critical disk writes pending in the system.



Important

In most cases a simple reboot will resolve boot volume corruption issues, so this is the first action you should take when troubleshooting this.

This topic describes how to determine if your Windows instance's boot volume is corrupted and what steps to take to troubleshoot and recover the corrupted boot volume. For Linux-based instances, see Recovering a Corrupted Boot Volume for Linux-Based Instances.

Detecting Boot Volume Corruption

When Windows operating systems detect boot volume corruption, the instance is usually able to recover from it by automatically repairing the file system. You can use a VNC console connection to verify that the instance isn't experiencing a system hang while repairing the file system, or to detect if there are other issues. VNC console connections enable you to see what's displayed through the VGA port, for more information about the VNC console, see Instance Console Connections.



Important

VNC console connections only work for virtual machine (VM) instances launched on October 13th, 2017 or later, and bare metal instances, launched on February 21, 2019 or later. If your instance does not support VNC console connections, proceed to Recovering the Boot Volume.

1. Create a VNC console connection for the instance.

   ### To create the VNC console connection for an instance

   a. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.

   b. In the list of instances, find the instance you think may have a corrupted boot volume, and then click the instance name.

   c. In the **Resources** section on the **Instance Details** page, click **Console Connections**, and then click **Create Console Connection**.

   d. Specify the public key portion for the SSH key, either by browsing and selecting a public key file, for example id_rsa.pub, or by pasting your public key into the text box, and then click **Create Console Connection**.

2. Connect to the instance through VNC console.

   ### To set up a secure tunnel to the VNC server on the instance using PowerShell on Windows

   a. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**. Click the instance you want to connect to.

   b. On the **Instances Details** page, in the **Resources** section, click **Console Connections**.

   c. Click the Actions icon (three dots), and then click **Connect with VNC**.

   d. Select **WINDOWS** for **PLATFORM**.

   e. Click **Copy** to copy the string to the clipboard.

   f. Paste the connection string copied from the previous step to Windows Powershell and hit enter to set up the secure connection.

   g. Once the connection has been established, open your VNC client and specify `localhost` as the host to connect to and `5900` as the port to use.

Check what is displayed in the VNC console to see if the instance is stuck in the boot process or if it is in the recovery partition.



Tip

For Windows 2012 and Windows 2016 operating systems, if the instance has booted into the recovery partition it may be possible to directly perform the steps to recover the boot volume in the recovery partition .

Detaching the Boot Volume

If you have detected that your instance's boot volume is corrupted, you need to detach the boot volume from the instance before you can begin troubleshooting and recovery steps.

1. Stop the instance. For more information, see Stopping and Starting an Instance.

   To stop the instance

   a. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
   b. In the list of instances, find the instance you have detected boot volume corruption for and then click the instance name to display the instance details.
   c. Click **Stop**.

2. Detach the boot volume from the instance. For more information, see Detaching a Boot Volume.

   To detach the boot volume from the instance

   a. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.

b. In the **Instances** list, select the instance you want to detach the boot volume from.
c. Click the name of the instance to display the instance details.
d. In the **Resources**, click **Boot Volume**.
e. Click the Actions icon (three dots), for the boot volume.
f. Click **Detach** and confirm the selection when prompted.

Recovering the Boot Volume

To troubleshoot and recover the corrupted boot volume, you need to attach the boot volume to a second instance as a data volume. For the second instance we recommend that you use an instance running an operating system that most closely matches the operating system for the boot volume's instance, and you should only attach boot volumes for Windows instances to other Windows instances. The second instance must be in the same availability domain and region as the boot volume's instance. If no existing instance is available create a new Windows instance using the steps described in Creating an Instance.

Once you have the second instance, make sure you can log in to the instance and that it is functional before proceeding with the recovery steps. After you have confirmed that the instance is functional perform the following steps.

1. Attach the boot volume to the second instance as a data volume. For more information, see Attaching a Volume.

To attach the boot volume as a data volume

a. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
b. In the **Instances** list, click the instance that you want to attach a volume to.
c. In the **Resources** section, click **Attached Block Volumes**.
d. Click **Attach Block Volume**.

e. Select **iSCSI** for the volume attachment type.
f. In the **Block Volume Compartment** drop-down list, select the compartment.
g. Choose the **SELECT VOLUME** option and then select the volume from the **Boot Volume** section of the **Block Volume** drop-down list.
h. Select **Read/Write** as the access type.
i. Click **Attach**.
When the volume's icon no longer lists it as **Attaching**, proceed with the next steps.

2. Connect to the second instance, see Connecting to a Windows Instance for more information.
3. Connect to the volume, see Connecting to a Volume on a Windows Instance for more information. Since you are attaching a boot volume as a data volume you must also run the `Connect-IscsiTarget` and set `IsMultiEnabled` to true. For example:

```
Set-Service -Name msiscsi -StartupType Automatic
Start-Service msiscsi
New-IscsiTargetPortal –TargetPortalAddress 169.254.2.4
Connect-IscsiTarget -NodeAddress iqn.2015-02.oracle.boot:uefi -TargetPortalAddress 169.254.2.4 -
IsPersistent $True -IsMultipathEnabled $True
```

4. Open **Computer Management** and navigate to **Storage**, and then **Disk Management**.
5. Select the new disk and mark it **Online**.
6. Click **This PC** and then right-click on the new disk and select **Properties**.
7. Navigate to **Tools**, **Error Checking**, and then **Check**.
8. Select **Scan Drive** and fix issues as they come up.
9. Mark the new disk **Offline**.
10. Open iscsi initiator with administrator privileges.
11. In **Favorite Targets**, remove the iscsi target of the attached volume.

Oracle-Provided Images

An image is a template of a virtual hard drive. The image determines the operating system and other software for an instance. The following table lists the images that are available in Oracle Cloud Infrastructure. For specific image and kernel version details, along with changes between versions, see Oracle-Provided Image Release Notes.

Image	Name	Description
Oracle Linux 7 Unbreakable Enterprise Kernel Release 5	Oracle-Linux-7.*x*-<*date*>-<*number*>	The Unbreakable Enterprise Kernel (UEK) is Oracle's optimized operating system kernel for demanding Oracle workloads. GPU shapes are supported with this image.
Oracle Linux 6 Unbreakable Enterprise Kernel Release 4	Oracle-Linux-6.*x*-<*date*>-<*number*>	The Unbreakable Enterprise Kernel (UEK) is Oracle's optimized operating system kernel for demanding Oracle workloads.
CentOS 7	CentOS-7-<*date*>-<*number*>	CentOS is a free, open-source Linux distribution that is suitable for use in enterprise cloud environments. For more information, see https://www.centos.org/.
CentOS 6	CentOS-6.*x*-<*date*>-<*number*>	CentOS is a free, open-source Linux distribution that is suitable for use in enterprise cloud environments. For more information, see https://www.centos.org/.

Image	Name	Description
Ubuntu 18.04 LTS	Canonical-Ubuntu-18.04-<*date*>-<*number*>	Ubuntu is a free, open-source Linux distribution that is suitable for use in the cloud. For more information, see https://www.ubuntu.com. Minimal Ubuntu is designed for automated use at scale. It uses a smaller boot volume, boots faster, and has a smaller surface for security patches than standard Ubuntu images. For more information, see https://wiki.ubuntu.com/Minimal. GPU shapes are supported with this image. You must install the appropriate GPU drivers from NVIDIA.
Ubuntu 16.04 LTS	Canonical-Ubuntu-16.04-<*date*>-<*number*>	Ubuntu is a free, open-source Linux distribution that is suitable for use in the cloud. For more information, see https://www.ubuntu.com. Minimal Ubuntu is designed for automated use at scale. It uses a smaller boot volume, boots faster, and has a smaller surface for security patches than standard Ubuntu images. For more information, see https://wiki.ubuntu.com/Minimal. GPU shapes are supported with this image. For Minimal Ubuntu, you must install the appropriate GPU drivers from NVIDIA.
Ubuntu 14.04 LTS	Canonical-Ubuntu-14.04-<*date*>-<*number*>	Ubuntu is a free, open-source Linux distribution that is suitable for use in the cloud. For more information, see https://www.ubuntu.com.

Image	Name	Description
Windows Server 2016	Windows-Server-2016-<*edition*>-Gen2.<*date*>-<*number*>	Windows Server 2016 supports running production Windows workloads on Oracle Cloud Infrastructure. GPU shapes are supported with this image. You must install the appropriate GPU drivers from NVIDIA.
Windows Server 2012 R2	Windows-Server-2012-R2-<*edition*>-<*gen*>-<*date*>-<*number*>	Windows Server 2012 R2 supports running production Windows workloads on Oracle Cloud Infrastructure. GPU shapes are supported with this image. You must install the GPU drivers from NVIDIA.
Windows Server 2008 R2 - Virtual Machine (VM)	Windows-Server-2008-R2-Enterprise-Edition-VM-<*date*>-<*number*>	Windows Server 2008 R2 Enterprise Edition supports running production Windows workloads on Oracle Cloud Infrastructure.

You also can create custom images of your boot disk OS and software configuration for launching new instances.

Essential Firewall Rules



Warning

Windows 2008 Server R2 images do not support restricting certain firewall rules for local principals, such as "Administrators", so any authenticated user on an instance can make outgoing connections to the iSCSI network endpoints (169.254.0.2:3260, 169.254.2.0/24:3260) that serve the instance's boot and block volumes.

All Oracle-provided images include rules that allow only "root" on Linux instances or "Administrators" on Windows Server 2012 R2 and Windows Server 2016 instances to make outgoing connections to the iSCSI network endpoints (169.254.0.2:3260, 169.254.2.0/24:3260) that serve the instance's boot and block volumes.

- Oracle recommends that you do not reconfigure the firewall on your instance to remove these rules. Removing these rules allows non-root users or non-administrators to access the instance’s boot disk volume.
- Oracle recommends that you do not create custom images without these rules unless you understand the security risks.
- Running Uncomplicated Firewall (UFW) on Ubuntu images may cause issues with these rules, so Oracle recommends that you do not enable UFW on your instances. See Ubuntu Instance fails to reboot after enabling Uncomplicated Firewall (UFW) for more information.

User Data

Oracle-provided images provide you with the ability to run custom scripts or provide custom metadata when the instance launches. To do this, you specify a custom startup script in the **Create Instance** dialog's **User Data** field. For more information about startup scripts, see cloud-init for Linux-based images and cloudbase-init for Windows-based images.

OS Updates for Linux Images

Oracle Linux and CentOS images are preconfigured to let you install and update packages from the repositories on the Oracle public yum server. The repository configuration file is in the `/etc/yum.repos.d` directory on your instance. You can install, update, and remove packages by using the yum utility.



Note

OS Security Updates for Oracle Linux and CentOS images

After you launch an instance using Oracle Linux or CentOS images, you are responsible for applying the required OS security updates published through the Oracle public yum server. For more information, see Installing and Using the Yum Security Plugin.

The Ubuntu image is preconfigured with suitable repositories to allow you to install, update, and remove packages.



Note

OS Security Updates for the Ubuntu image

After you launch an instance using the Ubuntu image, you are responsible for applying the required OS security updates using the `sudo apt-get upgrade` command.

Linux Kernel Updates

Oracle Linux images on Oracle Cloud Infrastructure include Oracle Linux Premier Support at no extra cost. This gives you all the services included with Premier Support, including Oracle

Ksplice. Ksplice enables you to apply important security and other critical kernel updates without a reboot. For more information, see About Oracle Ksplice and Ksplice Overview.

Ksplice is only available for Linux instances launched on or after February 15, 2017. For instances launched before August 25, 2017, you must install Ksplice before running it. See Installing and Running Oracle Ksplice for more information.



Note

Ksplice Support

Oracle Ksplice is not supported for CentOS and Ubuntu images, or on Linux images launched before February 15, 2017.

Configuring Automatic Package Updating on Instance Launch

You can configure your instance to automatically update to the latest package versions when the instance first launches using a cloud-init startup script. To do this, add the following code to the startup script:

```
package_upgrade: true
```

The upgrade process starts when the instance launches and runs in the background until it completes. To verify that it completed successfully, check the cloud-init logs in `/var/log`.

See User Data and Cloud config examples - Run apt or yum upgrade for more information.

Linux Image Details

See Lifetime Support Policy: Coverage for Oracle Linux and Oracle VM for details about the Oracle Linux support policy.

Users

For instances created using Oracle Linux and CentOS images, the user name `opc` is created automatically. The `opc` user has `sudo` privileges and is configured for remote access over the SSH v2 protocol using RSA keys. The SSH public keys that you specify while creating instances are added to the `/home/opc/.ssh/authorized_keys` file.

For instances created using the Ubuntu image, the user name `ubuntu` is created automatically. The `ubuntu` user has `sudo` privileges and is configured for remote access over the SSH v2 protocol using RSA keys. The SSH public keys that you specify while creating instances are added to the `/home/ubuntu/.ssh/authorized_keys` file.

Note that `root` login is disabled.

Remote Access

Access to the instance is permitted only over the SSH v2 protocol. All other remote access services are disabled.

Firewall Rules

Instances created using Oracle-provided images have a default set of firewall rules which allow only SSH access. Instance owners can modify those rules as needed, but must not restrict link local traffic to address 169.254.0.2 in accordance with the warning at the top of this page.

Be aware that Networking uses security lists to control packet-level traffic in and out of the instance. When troubleshooting access to an instance, make sure both the security lists associated with the instance's subnet and the instance's firewall rules are set correctly.

Cloud-init Compatibility

Instances created using Oracle-provided images are compatible with cloud-init. When launching an instance with the Core Services API, you can pass cloud-init directives with the metadata parameter. For more information, see LaunchInstance.

OCI Utilities

Instances created using Oracle Linux include a preinstalled set of utilities that are designed to make it easier to work with Oracle Linux images. These utilities consist of a service component and related command line tools.

The following table summarizes the components that are included in the OCI utilities.

Name	Description
`ocid`	The service component of oci-utils. This normally runs as a daemon started via `systemd`. This service scans for changes in the iSCSI and VNIC device configurations and caches the OCI metadata and public IP address of the instance.
`oci-iscsi-config`	Used to display and configure iSCSI devices attached to a compute instance. If no command line options are specified, lists devices that need attention.
`oci-metadata`	Displays metadata for the compute instance. If no command line options are specified, lists all available metadata. Metadata includes the instance OCID, display name, compartment, shape, region, availability domain, creation date, state, image, and any custom metadata that you provide, such as an SSH public key.
`oci-network-config`	Lists or configures Virtual Network Interface Cards (VNICs) attached to the compute instance. Lists the current Virtual Network Interface Cards (VNICs) provisioned in the cloud and configured on the instance. When a secondary VNIC is provisioned in the cloud, it must be explicitly configured on the instance using this script or similar commands.
`oci-public-ip`	Displays the public IP address of the current system in either human-readable or JSON format.

For more detailed information, see the OCI Utilities reference.

Windows OS Updates for Windows Images

Windows images include the Windows Update utility, which you can run to get the latest Windows updates from Microsoft. You have to configure the security list on the subnet on which the instance is running to allow instances to access Windows update servers.

Windows Image Details

Users

For instances created using Oracle-provided Windows images, the user name `opc` is created automatically. When you launch an instance using the Windows image, Oracle Cloud Infrastructure will generate an initial, one-time password that you can retrieve using the console or API. This password must be changed after you initially log on.

Remote Access

Access to the instance is permitted only through a Remote Desktop connection.

Firewall Rules

Instances created using the Windows image have a default set of firewall rules which allow Remote Desktop protocol or RDP access on port 3389. Instance owners can modify these rules as needed, but must not restrict link local traffic to 169.254.169.253 for the instance to activate with Microsoft Key Management Service (KMS). This is how the instance stays active and licensed.

Be aware that Networking uses security lists to control packet-level traffic in and out of the instance. When troubleshooting access to an instance, make sure both the security lists associated with the instance's subnet and the instance's firewall rules are set correctly.

User Data on Windows Images

On Windows images custom user data scripts are executed using cloudbase-init, which is the equivalent of cloud-init on Linux-based images. All Oracle-provided Windows images on Oracle Cloud Infrastructure include cloudbase-init installed by default. When an instance launches, cloudbase-init runs PowerShell, batch scripts, or additional user data content. See cloudbase-init Userdata for information about supported content types.

You can use user data scripts to perform various tasks, such as:

- Enable GPU support using a custom script to install the applicable GPU driver.
- Add or update local user accounts.
- Join the instance to a domain controller.
- Install certificates into the certificate store.
- Copy any required application workload files from the Object Storage service directly to the instance.



Warning

Do not include anything in the script that could trigger a reboot as this could impact the instance launch, causing it to fail. Any actions requiring a reboot should only be performed once the instance state is **RUNNING**.

Windows Remote Management

Windows Remote Management (WinRM) is enabled by default on Oracle-provided Windows images. WinRM provides you with the capability to remotely manage the operating system.

To use WinRM you need to add a stateful ingress rule for TCP traffic on destination port 5986.



Warning

Opening this port allows WinRM connections from public IP addresses. To only allow access from instances within the VCN ensure that this port is open on the appropriate security lists for the appropriate subnets. For more information, see Security Recommendations.

To enable WinRM access

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Click **Security Lists**, and then click the security list you're interested in.
4. Click **Edit All Rules**.
5. Under **Allow Rules for Ingress**, click **Add Rule**.
6. Enter the following for your new rule:
    a. **Source Type:** CIDR
    b. **Source CIDR:** 0.0.0.0/0
    c. **IP Protocol:** TCP
    d. **Source Port Range:** All
    e. **Destination Port Range:** 5986
7. When done, click **Save Security List Rules**.

To use WinRM on an instance

1. Get the instance's public IP address, see Getting the Instance Public IP Address and

Initial Windows Password.

2. Open Windows PowerShell on the Windows client you using to connect to the instance.
3. Run the following command:

```
# Get the public IP from your OCI running windows instance
$ComputerName = Public IP Address

# Store your username and password credentials (default username is opc)
$c = Get-Credential

# Options
$opt = New-PSSessionOption -SkipCACheck -SkipCNCheck -SkipRevocationCheck

# Create new PSSession (Pre-requisite: ensure security list has Ingress Rule for port 5986)
$PSSession = New-PSSession -ComputerName $ComputerName -UseSSL -SessionOption $opt -
Authentication Basic -Credential $c

# Connect to Instance PSSession
Enter-PSSession $PSSession

# To close connection use: Exit-PSSession
```

You can now remotely manage your Windows instance from your local PowerShell client.

Using NVIDIA GPU Cloud with Oracle Cloud Infrastructure

NVIDIA GPU Cloud (NGC) is a GPU-accelerated cloud platform optimized for deep learning and scientific computing. This topic provides an overview of how to use NGC with Oracle Cloud Infrastructure.

NVIDIA makes available on Oracle Cloud Infrastructure a customized Compute image optimized for the NVIDIA® Tesla Volta™ and Pascal™ GPUs . Running NGC containers on this instance provides optimum performance for deep learning jobs.

For those familiar with Oracle Cloud Infrastructure, to use NGC, you need to log into the Console, configure the settings as needed, and then create an instance based on the NGC image by specifying the image OCID. After launching the instance, you can SSH into the

instance and start running deep learning jobs using framework containers from the NGC container registry.

Prerequisites

- An Oracle Cloud Infrastructure tenancy. For more information, see Signing Up for Oracle Cloud Infrastructure.
- A cloud network to launch the instance into. For information about setting up cloud networks, see Using the Console in VCNs and Subnets.
- A key pair, to use for connecting to the instance via SSH. For information about generating a key pair, see Managing Key Pairs on Linux Instances.
- Security group and policy configured for the File Storage service. For more information, see Managing Groups, Getting Started with Policies, and Details for the File Storage Service.
- An NGC API key for authenticating with the NGC service.

To generate your NGC API key

1. Log into the NGC website.
2. On the NGC Registry page, click **Get API Key**.
3. Click **Generate API Key** and then click **Confirm** to generate the key. If you have an existing API key it will become invalid once you generate a new key.

Launching an instance based on the NGC image

USING THE CONSOLE

1. Open the Console, see Signing In to the Console for steps on how to do this.
2. Click **Compute**, choose a compartment you have permission to work in, and then click **Create Instance**.
3. In the **Create Instance** dialog box, specify the instance name, and select the availability domain for the instance.

4. Select **Image OCID** for **Boot Volume**.
5. Specify the image OCID applicable to your region as the **Image OCID**.

Region	Image OCID
us-ashburn-1	ocid1.image.oc1.iad.aaaaaaaaikn6ub6heefqxbe5fkiv4otbfe6ivza6y7di5khnkxkyvf2bkdta
eu-frankfurt-1	ocid1.image.oc1.eu-frankfurt-1.aaaaaaaauwuafl6uze6bnusphnn6y2mr5y7ajavx4kza7glyrqggxlnbo4zq

6. Select **Bare Metal Machine** for **Shape Type**.
7. Select the shape you want to use for **Shape**.
8. For **SSH Keys**, click **Choose SSH Key File**, navigate to the location where you saved the public key portion (.pub) of the SSH key file you created, select the file and click **Open**.
9. Select a virtual cloud network, and then click **Create Instance**.

You should now see the NGC instance with the status of **Provisioning**. Once the status has changed to **Running**, you can connect to the instance. For general information about launching Compute instances, see Creating an Instance.

See the following topics for accessing and working with the instance:

- Connecting to an Instance
- Stopping and Starting an Instance
- Terminating an Instance

When you connect to the instance using SSH you will be prompted for the NGC API key. If you supply the API key at the prompt, the instance will automatically log you into the NGC container registry so that you can run containers from the registry. You can choose not to supply the API key at the prompt and still log in to the instance. You can then log in later to the NGC container registry, see Logging in to the NGC Container Registry for more information.

USING THE CLI

Oracle Cloud Infrastructure provides a Command Line Interface (CLI) you can use to complete tasks. For more information, see Quickstart and Configuration. Use the launch command to create an instance, specifying image for **sourceType** and the applicable image OCID from the table below for **imageId** in **InstanceSourceDetails** for **LaunchInstanceDetails**.

Region	Image OCID
us-ashburn-1	ocid1.image.oc1.iad.aaaaaaaaikn6ub6heefqxbe5fkiv4otbfe6ivza6y7di5khnkxkyvf2bkdta
eu-frankfurt-1	ocid1.image.oc1.eu-frankfurt-1.aaaaaaaauwuafl6uze6bnusphnn6y2mr5y7ajavx4kza7glyrqggxlnbo4zq

To start and stop the instance with the CLI, use the action command. Use the terminate command to terminate the instance.

Using the File Storage Service for Persistent Data Storage

You can use the Overview of File Storage for data storage when working with NGC. See the following tasks for creating and working with the File Storage service:

- Creating File Systems
- Using the Console
- Using the API
- Managing File Systems
- Using the Command Line Interface (CLI)

Using the Block Volume Service for Persistent Data Storage

You can use the Block Volume service for data storage when working with NGC. For more information, see Overview of Block Volume. See the following tasks for creating and working

with the Block Volume service:

- Creating a Volume
- Attaching a Volume
- Connecting to a Volume

You can also use the CLI to manage block volumes, see the volume commands.

Examples of Running Containers

You first need to log into the NGC container registry. You can skip this section if you provided your API key when logging into the instance via SSH. If you did not provide your API key when connecting to your instance, then you must perform this step.

To log into the NGC container registry

1. Run the following Docker command:

```
docker login nvcr.io
```

2. When prompted for a username, enter `$oauthtoken`.
3. When prompted for a password enter your NGC API key.

At this point you can run Docker commands and access the NGC container registry from the instance.

Example: MNIST Training Run Using PyTorch Container

This sample demonstrates running the MNIST example under PyTorch. This example downloads the MNIST dataset from the web.

1. Pull and run the PyTorch container with the following Docker commands:

```
docker pull nvcr.io/nvidia/pytorch:17.10
nvidia-docker run --rm -it nvcr.io/nvidia/pytorch:17.10
```

2. Run the MNIST example with the following commands:

```
cd /opt/pytorch/examples/mnist
python main.py
```

Example: MNIST Training Run Using TensorFlow Container

This sample demonstrates running the MNIST example under TensorFlow. This example downloads the MNIST dataset from the web.

1. Pull and run the TensorFlow container with the following Docker commands:

```
nvcr.io/nvidia/tensorflow:17.10
nvidia-docker run --rm -it nvcr.io/nvidia/tensorflow:17.10
```

2. Run the MNIST_with_summaries example with the following commands:

```
cd /opt/tensorflow/tensorflow/examples/tutorials/mnist
python mnist_with_summaries.py
```

Compute Shapes

A shape is a template that determines the number of CPUs, amount of memory, and other resources allocated to a newly created instance. The following lists provide basic information about the available bare metal and VM shapes.

Bare Metal Shapes

Shape	Instance Type	OCPU	Memory (GB)	Local Disk	Network Bandwidth[1]	Max VNICs Total: Linux	Max VNICS Total: Windows
BM.Standard1.36[2]	Standard compute capacity	36	256	Block storage only	10 Gbps	36	1
BM.DenseIO1.36[2]	Dense I/O compute capacity	36	512	28.8 TB NVMe SSD (9 drives)	10 Gbps	36	1
BM.Standard2.52	X7-based standard compute capacity	52	768	Block storage only	2 x 25 Gbps	52 total (26 per physical NIC)	27 total (1 on the first physical NIC, 26 on the second)

Shape	Instance Type	OCPU	Memory (GB)	Local Disk	Network Bandwidth[1]	Max VNICs Total: Linux	Max VNICS Total: Windows
BM.DenseIO2.52	X7-based dense I/O compute capacity	52	768	51.2 TB NVMe SSD (8 drives)	2 x 25 Gbps	52 total (26 per physical NIC)	27 total (1 on the first physical NIC, 26 on the second)
BM.GPU2.2	X7-based GPU: 2xP100 NVIDIA GPUs	28	192	Block storage only	2 x 25 Gbps	28	15 (1 on the first physical NIC, 14 on the second)
BM.GPU3.8	X7-based GPU: 8xV100 NVIDIA GPUs	52	768	Block storage only	2 x 25 Gbps	52	27 (1 on the first physical NIC, 26 on the second)

Shape	Instance Type	OCPU	Memory (GB)	Local Disk	Network Bandwidth[1]	Max VNICs Total: Linux	Max VNICS Total: Windows
BM.Standard.E2.64	E1-based standard compute capacity: AMD CPUs	64	512	Block storage only	2 x 25 Gbps	75	76 (1 on the first physical NIC, 75 on the second)
BM.HPC2.36	X7-based high frequency compute capacity	36	384	6.7 TB NVMe SSD (1 drive)	1 x 25 Gbps 1 x 100 Gbps RDMA	50	1
1: Network bandwidth is based on expected bandwidth for traffic within a VCN. 2: X5-based shapes availability is limited to monthly universal credit customers existing on or before November 9th, 2018, in the us-phoenix-1, us-ashburn-1, and eu-frankfurt-1 regions.							

VM Shapes

Shape	OCPU	Memory (GB)	Local Disk (TB)	Network Bandwidth [1]	Max VNICs Total: Linux	Max VNICs Total: Windows
VM.Standard1.1[2]	1	7	Block Storage only	Up to 600 Mbps	2	1
VM.Standard1.2[2]	2	14	Block Storage only	Up to 1.2 Gbps	2	1
VM.Standard1.4[2]	4	28	Block Storage only	1.2 Gbps	4	1
VM.Standard1.8[2]	8	56	Block Storage only	2.4 Gbps	8	1
VM.Standard1.16[2]	16	112	Block Storage only	4.8 Gbps	16	1
VM.DenseIO1.4[2]	4	60	3.2 TB NVMe SSD	1.2 Gbps	4	1
VM.DenseIO1.8[2]	8	120	6.4 TB NVMe SSD	2.4 Gbps	8	1
VM.DenseIO1.16[2]	16	240	12.8 TB NVMe SSD	4.8 Gbps	16	1
VM.Standard2.1	1	15	Block Storage only	1 Gbps	2	2
VM.Standard2.2	2	30	Block Storage only	2 Gbps	2	2

Shape	OCPU	Memory (GB)	Local Disk (TB)	Network Bandwidth [1]	Max VNICs Total: Linux	Max VNICs Total: Windows
VM.Standard2.4	4	60	Block Storage only	4.1 Gbps	4	4
VM.Standard2.8	8	120	Block Storage only	8.2 Gbps	8	8
VM.Standard2.16	16	240	Block Storage only	16.4 Gbps	16	16
VM.Standard2.24	24	320	Block Storage only	24.6 Gbps	24	24
VM.Standard.E2.1	1	8	Block Storage only	700 Mbps	2	2
VM.Standard.E2.2	2	16	Block Storage only	1.4 Gbps	2	2
VM.Standard.E2.4	4	32	Block Storage only	2.8 Gbps	4	4
VM.Standard.E2.8	8	64	Block Storage only	5.6 Gbps	4	4
VM.DenseIO2.8	8	120	6.4 TB NVMe SSD	8.2 Gbps	8	8
VM.DenseIO2.16	16	240	12.8 TB NVMe SSD	16.4 Gbps	16	16

Shape	OCPU	Memory (GB)	Local Disk (TB)	Network Bandwidth [1]	Max VNICs Total: Linux	Max VNICs Total: Windows
VM.DenseIO2.24	24	320	25.6 TB NVMe SSD	24.6 Gbps	24	24
VM.GPU2.1 (GPU: 1xP100)	12	72	Block Storage only	8 Gbps	12	12
VM.GPU3.1 (GPU: 1xV100)	6	90	Block Storage only	4 Gbps	6	6
VM.GPU3.2 (GPU: 2xV100)	12	180	Block Storage only	8 Gbps	12	12
VM.GPU3.4 (GPU: 4xV100)	24	360	Block Storage only	24.6 Gbps	24	24

1: Network bandwidth is based on expected bandwidth for traffic within a VCN.

2: X5-based shapes availability is limited to monthly universal credit customers existing on or before November 9th, 2018, in the us-phoenix-1, us-ashburn-1, and eu-frankfurt-1 regions.

Installing and Running Oracle Ksplice

Oracle Ksplice enables you to apply important security and other critical kernel updates without a reboot. For more information, see About Oracle Ksplice and Ksplice Overview.

This topic describes how to install and configure Ksplice. Ksplice is only available for Oracle Linux instances launched on or after February 15, 2017. It is installed on instances launched on or after August 25, 2017, so you just need to run it on these instances to install the

available Ksplice patches. For instances launched prior to August 25, 2017, you must install it prior to running it.

Installing Ksplice on instances launched prior to August 25 2017

To install Ksplice you need to connect to your Linux instance by using a Secure Shell (SSH). See Connecting to an Instance for more information.

1. Use the following SSH command to access the instance.

   ```
   ssh -l opc@<public-ip-address>
   ```

   <public-ip-address> is your instance IP address that you retrieved from the Console, see Getting the Instance Public IP Address.

2. Run the following SSH commands to sudo to the root:

   ```
   sudo bash
   ```

3. Download the Ksplice installation script with the following SSH command:

   ```
   wget -N https://www.ksplice.com/uptrack/install-uptrack-oc
   ```

4. Once the script is downloaded, use the following SSH command to install Ksplice:

   ```
   sh install-uptrack-oc
   ```

Running Ksplice

To run Ksplice you need to connect to your Linux instance by using a Secure Shell (SSH). See Connecting to an Instance for more information.

1. Use the following SSH command to access the instance.

   ```
   ssh -l opc <public-ip-address>
   ```

   <public-ip-address> is your instance IP address that you retrieved from the Console, see Getting the Instance Public IP Address.

2. Run the following SSH commands to sudo to the root:

```
sudo bash
cd
```

3. To install available Ksplice patches, run the following SSH command:

```
uptrack-upgrade
```

Automatic Updates

You can configure automatic updates by setting the value of `autoinstall` to `yes` in `/etc/uptrack/uptrack.conf`.



Note

OS Security Updates for Oracle Linux images

Oracle Linux images are updated regularly with the necessary patches, but after you launch an instance using these images, you are responsible for applying the required OS security updates published through the Oracle public Yum server. For more information, see Installing and Using the Yum Security Plugin.

Managing Custom Images

Oracle Cloud Infrastructure uses images to launch instances. You specify an image to use when you launch an instance.

You can create a custom image of a bare metal instance's boot disk and use it to launch other instances. Instances you launch from your image include the customizations, configuration, and software installed when you created the image.

For details on Windows images, see Creating Windows Custom Images.

Custom images do not include the data from any attached block volumes. For information about backing up volumes, see Backing Up a Volume.



Tip

Oracle Cloud Infrastructure runs on Oracle's high-quality Sun servers. However, any hardware can experience a failure. Follow industry-wide hardware failure best practices to ensure the resilience of your solution. Some best practices include:

- Design your system with redundant compute nodes in different availability domains to support fail-over capability.
- Create a custom image of your system drive each time you change the image.
- Back up your data drives, or sync to spare drives, regularly.

If you experience a hardware failure and have followed these practices, you can terminate the failed instance, launch your custom image to create a new instance, and then apply the backup data.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let users launch Compute instances includes the ability to create and manage images. If the specified group doesn't need to launch instances or attach

volumes, you could simplify that policy to include only `manage instance-family`, and remove the statements involving `volume-family` and `virtual-network-family`.



Tip

When users create a custom image from an instance or launch an instance from a custom image, the instance and image don't have to be in the same compartment. However, users must have access to both compartments.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Limitations and Considerations

- Certain IP addresses are reserved for Oracle Cloud Infrastructure use and may not be used in your address numbering scheme. See IP Addresses Reserved for Use by Oracle for more information.
- Before you create a custom image of an instance, you must disconnect all iSCSI attachments and remove all iscsid node configurations from the instance. For steps, see Disconnecting From a Volume.
- When you create an image of a running instance, the instance shuts down and remains unavailable for several minutes. The instance restarts when the process completes.
- You cannot create additional custom images of an instance while the instance is engaged in the image creation process. When you start to create a custom image, the system implements a 20-minute timeout, during which you cannot create another image of the same instance. You can, however, create images of different instances at the same time.

- Custom images are available to all users authorized for the compartment in which the image was created.
- The maximum size for creating a custom image is 300 GB.
- The maximum size for importing a custom image is 300 GB.
- The maximum size for custom exported images is 50 GB.
- You can create a maximum of 25 custom images per region per root compartment.
- You cannot create an image of an Oracle Database instance.
- Windows custom images cannot be exported or downloaded.
- If you use a custom image and update the OS kernel on your instance, you must also upload the update to the network drive. See OS Kernel Updates for more information.

For information about how to deploy any version of any operating system that is supported by the Oracle Cloud Infrastructure hardware, see Bring Your Own Image (BYOI).

X5 and X7 Compatibility for Custom Images

Oracle X5 and X7 servers have different host hardware. As a result, using an X5 image on an X7 bare metal or virtual machine (VM) instance may not work without additional modifications. Oracle Cloud Infrastructure recommends for X7 hosts that you use the Oracle-provided images for X7. See Oracle-Provided Image Release Notes for more information about which images support X7. These images have been explicitly created and tested with the new hardware.

If you do attempt to use an existing X5 image on X7 hardware, note that CentOS 6 and all Windows versions are not cross-compatible.

Oracle Linux, Ubuntu 16.04, Ubuntu 14.04, and CentOS 7 are cross-compatible, however you need to update the kernel to the most recent version to install the latest device drivers. To do so, run the following commands from a terminal session:

- **Oracle Linux**

```
yum update
```

- **CentOS 7**

```
yum update
```

- **Ubuntu 16.04**

```
apt-get update
apt-get dist-upgrade
```

- **Ubuntu 14.04**

```
apt-get update
apt-get dist-upgrade
apt-get install linux-hwe-generic-trusty
```

The primary device drivers that are different between X5 and X7 hosts are:

- Network device drivers
- NVMe drive device drivers
- GPU device drivers

Additional updates may be required depending on how you have customized the image.

Using the Console

To access the Console, you must use a supported browser.

To create a custom image

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
2. Find the instance you want to use as the basis for an image.
3. Click the Actions icon (three dots), and then click **Create Custom Image**.
4. Enter a name for the image, and then click **Create Custom Image**. You can use the API to change the name later, if needed. You cannot use an Oracle-provided image name as a custom image name.



Note

Limits

If you see a message indicating that you are at the limit for custom images, you must delete at least one image before you can create another. Or, you can request a service limit increase.

To launch an instance from a custom image

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Custom Images**. Find the custom image you want to use.
2. Click the Actions icon (three dots), and then click **Create Instance**.
3. Provide additional launch options as described in Creating an Instance.

To edit custom image details

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Custom Images**.
2. Click the custom image you want to edit.
3. Click **Edit**
4. From the **Edit Image Details** dialog box, you can change the name, or add and remove compatible shapes for the custom image.
5. After you have finished editing the details, click **Save** to save your changes and close the dialog box.

To manage tags for a custom image

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Custom Images**.
2. Click the custom image you want to manage the tags for.
3. Click the **Tags** tab to view or edit the existing tags. Or click **Apply tag(s)** to add new ones.

For more information, see Resource Tags.

To delete a custom image

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Custom Images**.
2. Find the custom image you want to delete.
3. Click the Actions icon (three dots), and then click **Delete**.
4. Confirm when prompted.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operations to manage custom images:

- DeleteImage
- GetImage
- ListImages
- CreateImage
- UpdateImage

Creating Windows Custom Images

You can create a Windows custom image of a bare metal or VM instance's boot disk and use it to launch other instances. Instances you launch from your image include the customizations, configuration, and software installed when you created the image. For information about using the Console to manage custom images, see Managing Custom Images.

Windows supports two kinds of images: generalized and specialized. Generalized images are images that have been cleaned of instance-specific information. Specialized images are point-in-time snapshots of the boot disk of a running instance, and are useful for creating backups of an instance. Oracle Cloud Infrastructure supports bare metal and VM instances launched from both generalized and specialized custom Windows images.

Generalized images

A generalized image has a generalized OS disk, cleaned of computer-specific information. The images are generalized using Sysprep. Generalized images can be useful in scenarios such as quickly scaling an environment. Generalized images can be configured to preserve the existing `opc` user's account, including the password, at the time the image is created, or configured to recreate the `opc` user account, including generating a new, random password that you retrieve using the API. For background information, see Sysprep (Generalize) a Windows installation.

Specialized images

A specialized image has an OS disk that is already fully installed, and is essentially a copy of the original bare metal or VM instance. Specialized images are intended to be used for backups so that you can recover from a failure. Specialized images are useful when you are testing a task and may need to roll back to known good configuration. Specialized images are not recommended for cloning multiple identical Bare Metal instances or VMs in the same network because of issues with multiple computers having the same computer name and ID. When creating a specialized image, you must remember the `opc` user's password; a new password is not generated when the instance launches, and it cannot be retrieved from the console or API.

Creating a Generalized Image



Warning

Creating a generalized image from an instance will render the instance non-functional, so you should first create a custom image from the instance, and then launch a new instance from the custom image. Steps 1 - 2 describe how to do this. This is the instance that you'll generalize. Alternatively, you can make a backup image of the instance that you can use to launch a replacement instance if needed.



Warning

If you upgrade to PowerShell 5.0/WMF 5.0, you may encounter an issue where **Sysprep** fails which will prevent the image generalization process from completing. If this occurs, you may not be able to log into instances launched from the custom image. See Unable to log into instance launched from new Windows custom image for more information and how to workaround the issue.

1. Create the new image using To create a custom image.
2. Launch an instance from the new image using To launch an instance from a custom image.
3. Connect to the instance using a Remote Desktop client.
4. Go to Windows Generalized Image Support Files and download to the instance the file matching the Windows version for the instance.
5. Right-click the file, and then click **Run as administrator**.

6. Extract the files to **C:\Windows\Panther**. The following files are extracted into the Panther folder for all Windows Server versions:
   - Generalize.cmd
   - Specialize.cmd
   - unattend.xml
   - Post-Generalize.ps1

   For Windows Server 2008, the following file is also extracted into the Panther folder:
   - Windows2008-SnapshotUtilities.ps1
7. Optional: If you want to preserve the `opc` user account, edit `C:\Program Files\bmcs\imageType.json` and change the `imageType`setting to `custom`. A new password is not created and the password is not retrievable from the console or API. If you want to configure the generalized image to recreate the `opc` user account when a new instance is launched from the image, leave the `imageType` setting defaulted to `general`. The new account's password can be retrieved through the API using `GetWindowsInstanceInitialCredentials`.
8. Right-click **Generalize.cmd**, and then click **Run as administrator**. Keep in mind the following outcomes of running this command:
   - Your connection to the Remote Desktop client may immediately be turned off and you will be logged out of the instance. If this does not occur, you should log out of the instance yourself.
   - Because `sysprep generalize` turns off Remote Desktop, you won't be able to log in to the instance again.
   - Creating a generalized image essentially destroys the instance's functionality.

   You should wait for a few minutes before proceeding to step 9 to ensure the generalization process has completed.
9. Create the new image using To create a custom image.
10. After you create an image from an instance that has been generalized, Oracle recommends that you terminate that instance. Although it may appear to be running, it won't be fully operable.

Creating a Specialized Image



Important

When creating a specialized image, you must remember the `opc` user's password. It cannot be retrieved from the Console or API.

You create a specialized image the same way you create other custom images. For step-by-step instructions, see Managing Custom Images.

Image Import/Export

Oracle Cloud Infrastructure Compute enables you to share custom images across tenancies and regions using image import/export.

Linux-Based Operating Systems

The following Oracle Cloud Infrastructure operating systems support image import/export:

- Oracle Linux 6, Oracle Linux 7
- CentOS 6, CentOS 7
- Ubuntu 16.04, Ubuntu 14.04

For more information about these images, see Oracle-Provided Images.

Windows-Based Operating Systems

You can import custom Windows images. The following operating systems are supported:

- Windows Server 2008 R2 Standard, Enterprise, and Datacenter
- Windows Server 2012 Standard, Datacenter

- Windows Server 2012 R2 Standard, Datacenter
- Windows Server 2016 Standard, Datacenter

Verify Your Windows Operating System

When importing custom Windows images, you need to make sure that the version you select matches up with the Windows image that you imported. Failure to provide the correct version and SKU information could be a violation of your Microsoft Licensing Agreement.

Windows System Time Issue on Custom Windows Instances

If you change the time zone from the default setting on Windows VM instances, when the instance reboots or syncs with the hardware clock, the system time will revert back to the time for the default time zone. However, the time zone setting will stay set to the new time zone, so the system clock will be incorrect. You can fix this by setting the `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\TimeZoneInformation` registry key to 1.

OCI-provided Windows Images already have the `RealTimeIsUniversal` registry key set by default, but you will need to do this for any custom Windows images that you import.

To fix this issue for custom Windows images:

1. Open the Windows Registry Editor and navigate to the `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\TimeZoneInformation` registry key.
2. Create a new `DWORD` key named `RealTimeIsUniversal` and set the value to 1.
3. Reboot the instance.
4. Reset the time and time zone manually.

Exporting Windows Images

Image export is not supported for Windows images.

Bring Your Own Image Scenarios

You can also use image import/export to share custom images from Bring Your Own Image (BYOI) scenarios across tenancies and regions, so you don't need to re-create the image

manually in each region. You need to go through the steps required to manually create the image in one of the regions, but after this is done, you can export the image, making it available for import in additional tenants and regions. The exported image format is OCI, which is a TAR file that contains a QCOW2 file and Oracle Cloud Infrastructure-specific metadata.

Best practices for replicating an image across regions

You can use the Console or API to replicate an image from one region to other regions. At a high level:

1. Export the image to an Object Storage bucket in the same region as the image. For steps, see Exporting an Image.
2. Create a pre-authenticated request with read-only access for the exported image. For steps, see Working with Pre-Authenticated Requests.
3. Import the image to each region that you want to replicate it in. Use the pre-authenticated request URL as the Object Storage URL. For steps, see Importing an Image.

Object Storage Service URLs

When you import or export custom images using the Console, you may need to specify the Object Storage URL pointing to the location that you want to import the image from or export the image to. Object Storage URLs are structured as follows:

```
https://<host_name>/n/<namespace_name>/b/<bucket_name>/o/<object_name>
```

For example:

```
https://objectstorage.us-phoenix-1.oraclecloud.com/n/MyNamespace/b/MyBucket/o/MyCustomImage.qcow2
```

Pre-Authenticated Requests

When using import/export across tenants and regions you need to use an Object Storage pre-authenticated request. See Working with Pre-Authenticated Requests for instructions on creating a pre-authenticated request. When you go through these instructions, after you click

Create Pre-Authenticated Request, the **Pre-Authenticated Request Details** dialog opens. You need to make a copy of the pre-authenticated request URL displayed here, as this is the only time this will be displayed. This is the Object Storage URL you specify for import/export.



Note

Pre-authenticated requests for a bucket

With image export, if you create the pre-authenticated request for a bucket, you will need to append the object name to the generated URL. For example:

```
/o/MyCustomImage.qcow2
```

Exporting an Image

You can use the Console or API to export images, and the exported images are stored in the Oracle Cloud Infrastructure Object Storage service. To perform an image export, you need write access to the Object Storage bucket for the image. For more information, see Overview of Object Storage and Let users write objects to Object Storage buckets.

To export an image using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Custom Images**.
2. Find the custom image you want to export, click the Actions icon (three dots), and then click **Export Image**.
3. In the **Export Image** dialog box, specify the Object Storage location to export the image to. You have two options here: You can select a compartment and bucket, and then enter a name for the exported image, or you can enter the Object Storage URL.
4. Click **Export Image**.

After you click **Export Image**, the image status changes to **EXPORTING**. You can still launch instances while the image is exporting, but you can't delete the image until the export has finished. When the export is complete, the image status changes to **AVAILABLE**. If the image status changes to **AVAILABLE**, but you don't see the exported image in the Object Storage location you specified, this means that the export failed, and you will need to go through the steps again to export the image.

Importing an Image

You can use the Console or API to import exported images from Object Storage. To import an image, you need read access to the Object Storage object containing the image. For more information, see Let users download objects from Object Storage buckets.

To import an image using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Custom Images**.
2. Click **Import Image**.
3. Select the compartment name you want to import the image to.
4. Enter a name for the image.
5. Select the operating system.
6. Specify the Object Storage URL where the image is stored. When importing across regions or tenancies, you need to specify a pre-authenticated request URL.
7. Select the image type.
8. Select the launch mode.
    - For custom images where the image format is OCI, Oracle Cloud Infrastructure selects the applicable launch mode based on the launch mode for the source image.
    - For custom images exported from Oracle Cloud Infrastructure where the image type is QCOW2, select **Native Mode**.

- To import other custom images select **Paravirtualized Mode** or **Emulated Mode**. For more information, see Bring Your Own Image (BYOI).

9. Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
10. Click **Import Image**.

After you click **Import Image**, you'll see the imported image in the **Custom Images** list for the compartment, with a status of **IMPORTING**. When the import completes successfully, the status changes to **AVAILABLE**. If the status does not change, or no entry appears in the **Custom Images** list, the import failed. If the import failed, make sure you have read access to the Object Storage object, and that the object contains a supported image.

Editing Image Details

You can edit the details of custom images, such as the image name and compatible shapes for the image. For more information, see To edit custom image details in Managing Custom Images.

Managing Tags for an Image

You can apply tags to your resources, such images, to help you organize them according to your business needs. You can apply tags at the time you import an image, or you can update the image later with the desired tags.

To manage tags for an image

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Custom Images**.
2. Click the image you're interested in.

3. Click the **Tags** tab to view or edit the existing tags. Or click **Apply tag(s)** to add new ones.

For more information, see Resource Tags.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following API operations for custom image import/export:.

- ExportImage: Exports a custom image to Object Storage.
- CreateImage: To import an exported image, specify ImageSourceDetails in the request body.

X5 and X7 Compatibility for Image Import/Export

Oracle X5 and X7 servers have different host hardware. As a result, using an X5 image on an X7 bare metal or virtual machine (VM) instance may not work without additional modifications. Oracle Cloud Infrastructure recommends for X7 hosts that you use the Oracle-provided images for X7. See Oracle-Provided Image Release Notes for more information about which images support X7. These images have been explicitly created and tested with the new hardware.

If you do attempt to use an existing X5 image on X7 hardware, note that CentOS 6 and all Windows versions are not cross-compatible.

Oracle Linux, Ubuntu 16.04, Ubuntu 14.04, and CentOS 7 are cross-compatible, however you need to update the kernel to the most recent version to install the latest device drivers. To do so, run the following commands from a terminal session:

- **Oracle Linux**

```
yum update
```

- **CentOS 7**

```
yum update
```

- **Ubuntu 16.04**

```
apt-get update
apt-get dist-upgrade
```

- **Ubuntu 14.04**

```
apt-get update
apt-get dist-upgrade
apt-get install linux-hwe-generic-trusty
```

The primary device drivers that are different between X5 and X7 hosts are:

- Network device drivers
- NVMe drive device drivers
- GPU device drivers

Additional updates may be required depending on how you have customized the image.

Bring Your Own Image (BYOI)

The Bring Your Own Image (BYOI) feature enables you to bring your own versions of operating systems to the cloud as long as the underlying hardware supports it. The services do not depend on the OS you run.

The BYOI feature:

- Enables lift-and-shift cloud migration projects.
- Supports both old and new operating systems.
- Encourages experimentation.
- Increases infrastructure flexibility.



Note

Licensing Requirements

You must comply with all licensing requirements when you upload and start instances based on OS images that you supply.

Bringing Your Own Image

A critical part of any lift-and-shift cloud migration project is the migration of on-premises virtual machines (VMs) to the cloud. You can import your on-premises virtualized root volumes to Oracle Cloud Infrastructure using the custom image import feature, and then launch Compute instances using those images.

You can import Windows and Linux-based custom images.

- **Windows images**
  These Windows versions support custom image import:
  - Windows Server 2008 R2 Standard, Enterprise, Datacenter
  - Windows Server 2012 Standard, Datacenter
  - Windows Server 2012 R2 Standard, Datacenter
  - Windows Server 2016 Standard, Datacenter

  For steps to import a Windows image, see Importing Custom Windows Images.
- **Linux images**

These Linux distributions support custom image import:

Linux Distribution	Supported Versions	Preferred Launch Mode
RHEL	7 or later	Paravirtualized
	4.5, 5.9, 5.11, 6.9	Emulated
CentOS	7 or later	Paravirtualized
	4.0, 4.8, 5.11, 6.9	Emulated
Oracle Linux	7 or later	Paravirtualized
	4.5, 4.8, 5.8, 5.11, 6.2, 6.5	Emulated
Ubuntu	13.04 or later	Paravirtualized
	12.04	Emulated
FreeBSD	12 or later	Paravirtualized
	8, 9, 10, 11	Emulated
Debian	8 or later	Paravirtualized
	5.0.10, 6.0, 7	Emulated
SUSE	12.2 or later	Paravirtualized
	11, 12.1	Emulated

You might also have success importing other distributions of Linux.
For steps to import a Linux-based image, see Importing Custom Linux Images.

Bringing Your Own Hypervisor

- **Bring Your Own KVM:** You can bring your own operating system images or older operating systems such as Ubuntu 6.x, RHEL 3.x, CentOS 5.4 using KVM on bare metal instances. For more information, see Getting Started: Oracle Linux KVM Image for Oracle Cloud Infrastructure and Installing and Configuring KVM on Bare Metal Instances with Multi-VNIC.
- **Bring your own OVM:** You can bring your Oracle VM workload to Oracle Cloud Infrastructure. For more information, see Installing and Configuring Oracle VM on Oracle Cloud Infrastructure.
- **Bring your own Hyper-V:** You can bring your own operating system images or older operating systems such as Windows Server 2003, Windows Server 2008, and older Linux -based operating systems using Hyper-V on bare metal instances. For a full list of supported Hyper-V guests, see Supported Windows guest operating systems for Hyper-V on Windows Server and Supported Linux and FreeBSD virtual machines for Hyper-V on Windows. See Deploying Hyper-V on Oracle Cloud Infrastructure for more information.

Importing Custom Windows Images

The Compute service enables you to import Windows images that were created outside of Oracle Cloud Infrastructure. For example, you can import images running on your on-premises physical or virtual machines (VMs), or VMs running in Oracle Cloud Infrastructure Classic. You can then launch your imported images on Compute virtual machines.

Supported Operating Systems

These Windows versions support custom image import:

- Windows Server 2008 R2 Standard, Enterprise, Datacenter
- Windows Server 2012 Standard, Datacenter
- Windows Server 2012 R2 Standard, Datacenter
- Windows Server 2016 Standard, Datacenter



Note

- Oracle Cloud Infrastructure has tested the operating systems listed previously and will support customers in ensuring that instances launched from these images and built according to the guidelines in this topic are accessible using RDP.
- For OS editions not listed previously, Oracle Cloud Infrastructure will provide commercially reasonable support to customers in an effort to get instances that are launched from these images accessible via RDP.
- Support from Oracle Cloud Infrastructure in launching an instance from a custom OS does not ensure that the operating system vendor also supports the instance.

Windows Source Image Requirements

Custom images must meet the following requirements:

- The maximum image size is 300 GB.
- The image must be set up for BIOS boot.
- Only one disk is supported, and it must be the boot drive with a valid master boot record (MBR) and boot loader. You can migrate additional data volumes after you import the image's boot volume.
- The boot process must not require additional data volumes to be present for a successful boot.
- The disk image cannot be encrypted.

- The disk image must be a VMDK or QCOW2 file.
    - Create the image file by cloning the source volume, not by creating a snapshot.
    - VMDK files must be either the "single growable" (monolithicSparse) type or the "stream optimized" (streamOptimized) type, both of which consist of a single VMDK file. All other VMDK formats, such as those that use multiple files, split volumes, or contain snapshots, are not supported.
- The network interface must use DHCP to discover the network settings. When you import a custom image, existing network interfaces are not recreated. Any existing network interfaces are replaced with a single NIC after the import process is complete. You can attach additional VNICs after you launch the imported instance.
- The network configuration must not hardcode the MAC address for the network interface.

Preparing Windows VMs for Import

Before you can import a custom Windows image, you must prepare the image to ensure that instances launched from the image can boot correctly and that network connections will work.

You can perform the tasks described in this section on the running source system. If you have concerns about modifying the live source system, you can export the image as-is, import it into Oracle Cloud Infrastructure, and then launch an instance based on the custom image. You can then connect to the instance using the VNC console and perform the preparation steps.



Important

The system drive where Windows is installed will be imported to Oracle Cloud Infrastructure. All partitions on the drive will follow through the imported image. Any other drives will not be imported and you must re-create them on the instance after import. You will then need to manually move the data on the non-system drives.

To prepare a Windows VM for import:

1. Follow your organization's security guidelines to ensure that the Windows system is secured. This can include, but is not limited to the following tasks:
    - Install the latest security updates for the operating system and installed applications.
    - Enable the firewall, and configure it so that you only enable the rules which are needed.
    - Disable unnecessary privileged accounts.
    - Use strong passwords for all accounts.
2. Configure Remote Desktop Protocol (RDP) access to the image:
    a. Enable Remote Desktop connections to the image.
    b. Modify the Windows Firewall inbound port rule to allow RDP access for both Private and Public network location types. When you import the image, the Windows Network Location Awareness service will identify the network connection as a Public network type.
3. Determine whether the current Windows license type is a volume license by running the following command in PowerShell:

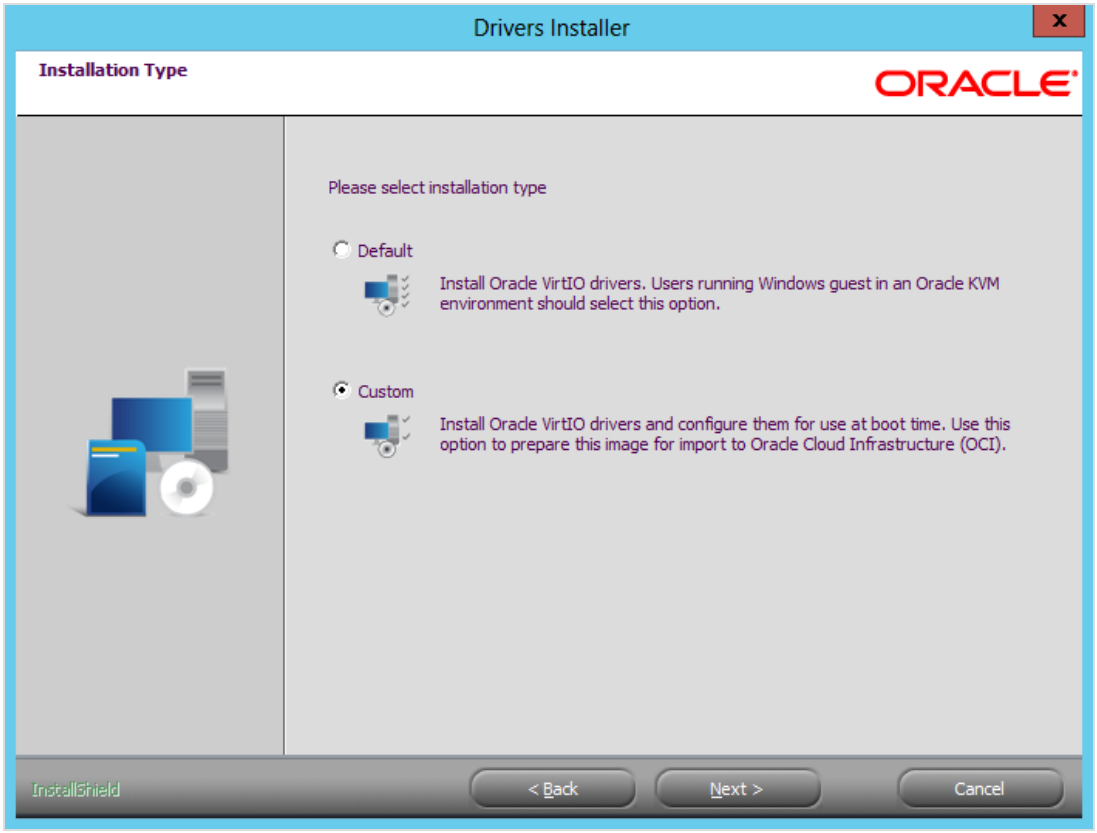
    ```
    Get-CimInstance -ClassName SoftwareLicensingProduct | where {$_.PartialProductKey} | select ProductKeyChannel
    ```

    If the license is not a volume license, after you import the image, you will update the license type.
4. If you plan to launch the imported image on more than one VM instance, create a generalized image of the boot disk. A generalized image is cleaned of computer-specific information, such as unique identifiers. When you create instances from a generalized image, the unique identifiers are regenerated. This prevents two instances that are created from the same image from colliding on the same identifiers.
5. Create a backup of the root volume.
6. If the VM has remotely attached storage, such as NFS or block volumes, configure any services that rely on this storage to start manually. Remotely attached storage is not

available the first time that an imported instance boots on Oracle Cloud Infrastructure.

7. Ensure that all network interfaces use DHCP, and that the MAC address and IP addresses are not hardcoded. See your system documentation for steps to perform network configuration for your system.
8. Download the Oracle Windows VirtIO drivers:
    a. Log in to the Oracle Software Delivery Cloud site.
    b. In the **All Categories** list, select **Release**.
    c. Type **Oracle Linux 7.6** in the search box and click **Search**.
    d. Add **REL: Oracle Linux 7.6.x** to your cart, and then click **Checkout**.
    e. In the **Platforms/Languages** list, select **x86 64 bit**. Click **Continue**.
    f. Accept the license agreement and then click **Continue**.
    g. Select the check box next to **Oracle VirtIO Drivers Version for Microsoft Windows 1.1.3**. Clear the other check boxes.
    h. Click **Download** and then follow the prompts.

9. Install the Oracle VirtIO drivers for Windows:
    a. Follow the prompts in the installation wizard. On the **Installation Type** page, select **Custom**, as shown in the following screenshot.



    b. Reboot the VM.
10. Stop the VM.
11. Clone the stopped VM as a VMDK or QCOW2 file, and then export the image from your virtualization environment. See the tools documentation for your virtualization environment for steps.

Importing a Windows-Based VM

After you prepare a Windows image for import, follow these steps to import the image:

1. Upload the VMDK or QCOW2 file to Oracle Cloud Infrastructure:
   a. Upload the file to an Object Storage bucket. You can upload the file using the Console or using the command line interface (CLI). If you use the CLI, use the following command:

      ```
      oci os object put -bn <destination_bucket_name> --file <path_to_the_VMDK_or_QCOW2_file>
      ```

   b. Copy the URL of the file that you uploaded: On the **Bucket Details** page, click the the Actions icon (three dots) next to the file, and then click **Details**. Copy the **URL Path (URI)**.
2. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Custom Images**.
3. Click **Import Image**.
4. In the **Create in Compartment** list, select the compartment that you want to import the image to.
5. Enter a name for the image.
6. For the **Operating System**, select **Windows**.
7. In the **Operating System Version** list, select the version of Windows.
8. Confirm that you chose the operating system version that complies with your Microsoft licensing agreement, and then select the compliance check box.

   Important

   Failure to provide the correct version and SKU information could be a violation of your Microsoft Licensing Agreement.
9. In the **Object Storage URL** field, paste the URL of the file that you uploaded.

10. For the **Image Type**, select the file type of the image, either **VMDK** or **QCOW2**.
11. In the **Launch Mode** area, select **Paravirtualized Mode**.
12. **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
13. Click **Import Image**.
    The imported image appears in the **Custom Images** list for the compartment, with a status of **IMPORTING**. When the import completes successfully, the status changes to **AVAILABLE**.
    If the status doesn't change, or no entry appears in the **Custom Images** list, the import failed. Ensure that you have read access to the Object Storage object, and that the object contains a supported image.
14. Complete the post-import tasks.

Post-Import Tasks for Windows Images

After you import a custom Windows-based image, do the following:

1. Create an instance based on the custom image. For the image source, select **Custom Images**, and then select the image that you imported.
2. Enable Remote Desktop Protocol (RDP) access to the Compute instance.
3. Connect to the instance using RDP.
4. If the instance requires any remotely attached storage, such as block volumes or file storage, create and attach it.
5. Create and attach any required secondary VNICs.
6. Test that all applications are working as expected.
7. Reset any services that were set to start manually.

8. Register the instance with the Oracle-provided Key Management Service (KMS) server:
    a. On the instance, open PowerShell as Administrator.
    b. To set the KMS endpoint, run the following command:

```
slmgr /skms 169.254.169.253:1688
```

    c. If the Windows license type that you noted while preparing the image isn't a volume license, you must update the license type. Run the following command:

```
slmgr /ipk <setup key>
```

    <setup key> is the KMS client setup key that corresponds to the version of Windows that you imported:

Windows Version	KMS Client Setup Key
Windows Server 2008 R2 Standard	YC6KT-GKW9T-YTKYR-T4X34-R7VHC
Windows Server 2008 R2 Enterprise	489J6-VHDMP-X63PK-3K798-CPX3Y
Windows Server 2008 R2 Datacenter	74YFP-3QFB3-KQT8W-PMXWJ-7M648
Windows Server 2012 Standard	XC9B7-NBPP2-83J2H-RHMBY-92BT4
Windows Server 2012 Datacenter	48HP8-DN98B-MYWDG-T2DCC-8W83P
Windows Server 2012 R2 Standard	D2N9P-3P6X9-2R39C-7RTCD-MDVJX
Windows Server 2012 R2 Datacenter	W3GGN-FT8W3-Y4M27-J84CP-Q3VJ9
Windows Server 2016 Standard	WC2BQ-8NRM3-FDDYY-2BFGV-KHKQY
Windows Server 2016 Datacenter	CB7KF-BWN84-R7R2Y-793K2-8XDDG

    d. To activate Windows, run the following command:

```
slmgr /ato
```

e. To verify the license status, run the following command:

```
Get-CimInstance -ClassName SoftwareLicensingProduct | where {$_.PartialProductKey} |
select Description, LicenseStatus
```

If the `LicenseStatus` is `1`, the instance is properly licensed.

Importing Custom Linux Images

The Compute service enables you to import Linux-based images that were created outside of Oracle Cloud Infrastructure. For example, you can import images running on your on-premises physical or virtual machines (VMs), or VMs running in Oracle Cloud Infrastructure Classic. You can then launch your imported images on Compute virtual machines.

Supported Operating Systems

You can launch imported Linux VMs in either paravirtualized mode or emulated mode.

Paravirtualized mode offers better performance than emulated mode. We recommend that you use paravirtualized mode if your OS supports it. Linux-based operating systems running the kernel version 3.4 or later support paravirtualized drivers. You can verify your system's kernel version using the uname command.

To verify the kernel version using the uname command

Run the following command:

```
uname -a
```

The output should look similar to this sample:

```
Linux ip_bash 4.14.35-1818.2.1.el7uek.x86_64 #2 SMP Mon Aug 27 21:16:31 PDT 2018 x86_64 x86_64 x86_64
GNU/Linux
```

The kernel version is the number at the first part of output string. In the sample output shown previously, the version is 4.14.35.

These Linux distributions support custom image import:

Linux Distribution	Supported Versions	Preferred Launch Mode
RHEL	7 or later	Paravirtualized
	4.5, 5.9, 5.11, 6.9	Emulated
CentOS	7 or later	Paravirtualized
	4.0, 4.8, 5.11, 6.9	Emulated
Oracle Linux	7 or later	Paravirtualized
	4.5, 4.8, 5.8, 5.11, 6.2, 6.5	Emulated
Ubuntu	13.04 or later	Paravirtualized
	12.04	Emulated
FreeBSD	12 or later	Paravirtualized
	8, 9, 10, 11	Emulated
Debian	8 or later	Paravirtualized
	5.0.10, 6.0, 7	Emulated
SUSE	12.2 or later	Paravirtualized
	11, 12.1	Emulated



Note

- Oracle Cloud Infrastructure has tested the operating systems listed in the previous table and will support customers in ensuring that instances launched from these images and built according to the guidelines in this topic are accessible using SSH.
- For OS versions not listed in the previous table, Oracle Cloud Infrastructure will provide commercially reasonable support to customers in an effort to get instances that are launched from these images accessible via SSH.
- Support from Oracle Cloud Infrastructure in launching an instance from a custom OS does not ensure that the operating system vendor also supports the instance. Customers running Oracle Linux on Oracle Cloud Infrastructure automatically have access to Oracle Linux Premier Support.



Tip

If your image supports paravirtualized drivers, you can convert your existing emulated mode instances into paravirtualized instances. Create a custom image of your instance, export it to Object Storage, and then reimport it using paravirtualized mode.

Linux Source Image Requirements

Custom images must meet the following requirements:

- The maximum image size is 300 GB.
- The image must be set up for BIOS boot.
- Only one disk is supported, and it must be the boot drive with a valid master boot record (MBR) and boot loader. You can migrate additional data volumes after you import the image's boot volume.
- The boot process must not require additional data volumes to be present for a successful boot.
- The boot loader should use LVM or a UUID to locate the boot volume.
- The disk image cannot be encrypted.
- The disk image must be a VMDK or QCOW2 file.
    - Create the image file by cloning the source volume, not by creating a snapshot.
    - VMDK files must be either the "single growable" (monolithicSparse) type or the "stream optimized" (streamOptimized) type, both of which consist of a single VMDK file. All other VMDK formats, such as those that use multiple files, split volumes, or contain snapshots, are not supported.
- The network interface must use DHCP to discover the network settings. When you import a custom image, existing network interfaces are not recreated. Any existing network interfaces are replaced with a single NIC after the import process is complete. You can attach additional VNICs after you launch the imported instance.
- The network configuration must not hardcode the MAC address for the network interface.

We recommend that you enable certificate-based SSH, however this is optional. If you want your image to automatically use SSH keys supplied from the **User Data** field when you launch an instance, you can install cloud-init when preparing the image. See Creating an Instance for more information about the **User Data** field.

Preparing Linux VMs for Import

Before you import a custom Linux image, you must prepare the image to ensure that instances launched from the image can boot correctly and that network connections will work. Do the following:

1. Optionally, configure your Linux image to support serial console connections. A console connection can help you remotely troubleshoot malfunctioning instances, such as an imported image that does not complete a successful boot.
2. Create a backup of the root volume.
3. If the VM has remotely attached storage, such as NFS or block volumes, configure any services that rely on this storage to start manually. Remotely attached storage is not available the first time that an imported instance boots on Oracle Cloud Infrastructure.
4. Ensure that all network interfaces use DHCP, and that the MAC address and IP addresses are not hardcoded. See your system documentation for steps to perform network configuration for your system.
5. Stop the VM.
6. Clone the stopped VM as a VMDK or QCOW2 file, and then export the image from your virtualization environment. See the tools documentation for your virtualization environment for steps.

Importing a Linux-Based VM

After you prepare a Linux image for import, follow these steps to import the image:

1. Upload the VMDK or QCOW2 file to Oracle Cloud Infrastructure:
   a. Upload the file to an Object Storage bucket. You can upload the file using the Console or using the command line interface (CLI). If you use the CLI, use the following command:

```
oci os object put -bn <destination_bucket_name> --file <path_to_the_VMDK_or_QCOW2_file>
```

b. Copy the URL of the file that you uploaded: On the **Bucket Details** page, click the the Actions icon (three dots) next to the file, and then click **Details**. Copy the **URL Path (URI)**.

2. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Custom Images**.
3. Click **Import Image**.
4. In the **Create in Compartment** list, select the compartment that you want to import the image to.
5. Enter a name for the image.
6. For the **Operating System**, select **Linux**.
7. In the **Object Storage URL** field, paste the URL of the file that you uploaded.
8. For the **Image Type**, select the file type of the image, either **VMDK** or **QCOW2**.
9. Depending on your image's version of Linux, in the **Launch Mode** area, select **Paravirtualized Mode** or **Emulated Mode**. If your image supports paravirtualized drivers, we recommend that you select paravirtualized mode.
10. **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
11. Click **Import Image**.

    The imported image appears in the **Custom Images** list for the compartment, with a status of **IMPORTING**. When the import completes successfully, the status changes to **AVAILABLE**.

    If the status doesn't change, or no entry appears in the **Custom Images** list, the import failed. Ensure that you have read access to the Object Storage object, and that the object contains a supported image.
12. Complete the post-import tasks.

Post-Import Tasks for Linux Images

After you import a custom Linux-based image, do the following:

1. Create an instance based on the custom image. For the image source, select **Custom Images**, and then select the image that you imported.
2. Connect to the instance using SSH.
3. If the instance requires any remotely attached storage, such as block volumes or file storage, create and attach it.
4. Create and attach any required secondary VNICs.
5. Test that all applications are working as expected.
6. Reset any services that were set to start manually.
7. If you enabled serial console access to the image, test it by creating a serial console connection to the instance.

See the current issues and workarounds for known issues with imported custom images.

Enabling Serial Console Access for Imported Linux Images

You can configure your custom Linux image to support connections using the serial console feature in the Compute service.

For more information about serial console connections, and steps to troubleshoot if your image has network connectivity issues after it is launched, see Instance Console Connections.

The serial console connection in Oracle Cloud Infrastructure uses the first serial port, ttyS0, on the VM. The boot loader and the operating system should be configured to use ttyS0 as a console terminal for both input and output.

Configuring the Boot Loader

The steps to configure the boot loader to use ttyS0 as a console terminal for both input and output depend on the GRUB version. Run the following command on the operating system to determine the GRUB version:

```
grub install --version
```

If the version number returned is 2.x, use the steps for GRUB 2. For earlier versions, use the steps for GRUB.

To configure GRUB2

1. Run the following command to modify the GRUB configuration file:

   ```
   sudo vi /etc/default/grub
   ```

2. Confirm that the configuration file contains the following:

   ```
   GRUB_SERIAL_COMMAND="serial --unit=0 --speed=115200"
                                                    GRUB_TERMINAL="serial console"
   ```

3. Append the following to the end of the `GRUB_CMDLINE_LINUX` line:

   ```
   console=tty1 console=ttyS0,115200
   ```

   If `GRUB_CMDLINE_LINUX` does not exist, create this line, using `GRUB_CMDLINE_OUTPUT` as a template.

4. Regenerate the GRUB2 configuration using the following command:

   ```
   sudo grub2-mkconfig -o /boot/grub2/grub.cfg
   ```

   If you have a beta version of GRUB 2, use this command instead:

   ```
   sudo grub-mkconfig -o /boot/grub/grub.cfg
   ```

To configure GRUB

1. Run the following command to modify the GRUB configuration file:

   ```
   sudo vi /boot/grub/grub.conf
   ```

2. Add following after the line containing `timeout`:

   ```
   serial --unit=0 --speed=115200
   terminal --timeout=5 serial console
   ```

3. Append the following to each `kernel` line:

```
console=tty1 console=ttyS0,115200
```

CONFIGURING THE OPERATING SYSTEM

The operating system may already be configured to use ttyS0 as a console terminal for both input and output. To verify, run the following command:

```
sudo vi /etc/securetty
```

Check the file for `ttyS0`. If you don't see it, append `ttyS0` to the end of the file.

VALIDATING SERIAL CONSOLE ACCESS

After completing the steps to enable serial console access to the image, you should validate that serial console access is working by testing the image with serial console in your virtualization environment. Consult the documentation for your virtualization environment for steps to do this. Verify that the boot output displays in the serial console output and that there is interactive input after the image has booted.

TROUBLESHOOTING THE SERIAL CONSOLE

If no output is displayed on the serial console, verify in the configuration for your virtualization environment that the serial console device is attached to the first serial port.

If the serial console displays output, but there is no interactive input available, check that there is a terminal process listening on the ttyS0 port. To do this, run the following command:

```
ps aux | grep ttyS0
```

This command should output a terminal process that is listening on the ttyS0 port. For example, if your system is using getty, you will see the following output:

```
/sbin/getty ttyS0
```

If you don't see this output, it is likely that a login process is not configured for the serial console connection. To resolve this, enable the init settings, so that a terminal process is listening on the ttyS0 at startup.

For example, if your system is using getty, add the following command to the init settings to run on system startup:

```
getty -L 9600 ttyS0 vt102
```

The steps to do this will vary depending on the operating system, so consult the documentation for the image's operating system.

OS Kernel Updates



Note

This topic applies only to Linux instances that were launched before February 15, 2017. Linux instances launched on or after February 15, 2017 boot directly from the image and do not require further action for kernel updates.

Oracle Cloud Infrastructure boots each instance from a network drive. This configuration requires additional actions when you update the OS kernel.

Oracle Cloud Infrastructure uses Unified Extensible Firmware Interface (UEFI) firmware and a Preboot eXecution Environment (PXE) interface on the host server to load iPXE from a Trivial File Transfer Protocol (TFTP) server. The iPXE implementation runs a script to boot Oracle Linux. During the boot process, the system downloads the kernel, the `initrd` file, and the kernel boot parameters from the network. The instance does not use the host's GRUB boot loader.

Normally, the `yum update kernel-uek` command edits the GRUB configuration file, either `grub.cfg` or `grub.conf`, to configure the next boot. Since bare metal instances do not use the GRUB boot loader, changes to the GRUB configuration file are not implemented. When you update the kernel on your instance, you also must upload the update to the network to ensure a successful boot process. The following approaches address this need:

- Instances launched from an Oracle-provided image include an Oracle yum plug-in that seamlessly handles the upload when you run the `yum update kernel-uek` command.
- If you use a custom image based on an Oracle-provided image, the included yum plug-in will continue to work, barring extraordinary changes.
- If you install your own package manager, you must either write your own plug-in or upload the kernel, initrd, and kernel boot parameters manually.

Oracle Yum Plug-in

On instances launched with an Oracle-provided image, you can find the Oracle yum plug-in at:

```
/usr/share/yum-plugins/kernel-update-handler.py
```

The plug-in configuration is at:

```
/etc/yum/pluginconf.d
```

The plug-in looks for two variables in the `/etc/sysconfig/kernel` file, `UPDATEDEFAULT` and `DEFAULTKERNEL`. It picks up the updates only when the first variable is set to "yes" and the `DEFAULTKERNEL` value matches the kernel being updated. For example:

```
# UPDATEDEFAULT specifies if new-kernel-pkg should make
# new kernels the default
UPDATEDEFAULT=yes

# DEFAULTKERNEL specifies the default kernel package type
DEFAULTKERNEL=kernel-uek
```

Oracle-provided images incorporate the Unbreakable Enterprise Kernel (UEK). If you want to switch to a non-UEK kernel, you must update the `DEFAULTKERNEL` value to "kernel" before you run `yum update kernel`.

Manual Updates



Tip

Oracle recommends using the Oracle yum plug-in to update the kernel.

If you manually upload the updates, there are four relevant URLs:

`http://169.254.0.3/kernel`

`http://169.254.0.3/initrd`

`http://169.254.0.3/cmdline`

`http://169.254.0.3/activate`

The first three URLs are for uploading files (HTTP request type PUT). The fourth URL is for activating the uploaded files (HTTP request type POST). The system discards the uploaded files if they are not activated before the host restarts.

The kernel and initrd are simple file uploads. The cmdline upload must contain the kernel boot parameters found in the `grub.cfg` or `grub.conf` file, depending on the Linux version. The following example is an entry from the `/boot/efi/EFI/redhat/grub.cfg` file in Red Hat Linux 7. The highlighted text represents the parameters to upload.

```
kernel /boot/vmlinuz-4.1.12-37.5.1.el6uek.x86_64 ro root=UUID=8079e287-53d7-4b3d-b708-c519cf6829c8 rd_
NO_LUKS  KEYBOARDTYPE=pc KEYTABLE=us netroot=iscsi:@169.254.0.2::3260:iface1:eth0::iqn.2015-
02.oracle.boot:uefi rd_NO_MD SYSFONT=latarcyrheb-sun16 ifname=eth0:90:e2:ba:a2:e3:80 crashkernel=auto
iscsi_initiator=iqn.2015-02. rd_NO_LVM ip=eth0:dhcp rd_NO_DM LANG=en_US.UTF-8 console=tty0
console=ttyS0,9600 iommu=on
```

The following command returns what is being uploaded to the cmdline file.

```
cat /tmp/cmdline
```

A typical response resembles the following.

```
ro root=UUID=8079e287-53d7-4b3d-b708-c519cf6829c8 rd_NO_LUKS  KEYBOARDTYPE=pc KEYTABLE=us
netroot=iscsi:@169.254.0.2::3260:iface1:eth0::iqn.2015-02.oracle.boot:uefi rd_NO_MD SYSFONT=latarcyrheb-
```

```
sun16 ifname=eth0:90:e2:ba:a2:e3:80 crashkernel=auto iscsi_initiator=iqn.2015-02. rd_NO_LVM ip=eth0:dhcp
rd_NO_DM LANG=en_US.UTF-8 console=tty0 console=ttyS0,9600 iommu=on
```

The following commands update the `cmdline` and `initrd` files, and then activate the changes.

```
CKSUM=`md5sum /tmp/cmdline | cut -d ' ' -f 1`
```

```
sudo curl -X PUT --data-binary @/tmp/cmdline -H "Content-MD5: $CKSUM" http://169.254.0.3/cmdline
```

```
CKSUM=`md5sum /boot/initramfs-3.8.13-118.8.1.el7uek.x86_64.img | cut -d ' ' -f 1`
```

```
sudo curl -X PUT --data-binary @/boot/initramfs-3.8.13-118.8.1.el7uek.x86_64.img -H "Content-MD5:
$CKSUM" http://169.254.0.3/initrd
```

```
sudo curl -X POST http://169.254.0.3/activate
```

Managing Key Pairs on Linux Instances

Instances launched using Oracle Linux, CentOS, or Ubuntu images use an SSH key pair instead of a password to authenticate a remote user (see Security Credentials). A key pair consists of a private key and public key. You keep the private key on your computer and provide the public key every time you launch an instance.

When you connect to an instance using SSH, you provide the path to the key pair file in the SSH command. You can have as many key pairs as you want, or you can keep it simple and use one key pair for all or several of your instances.

To create key pairs, you can use a third-party tool such as OpenSSH on UNIX-style systems (including Linux, Solaris, BSD, and OS X) or PuTTY Key Generator on Windows.

Prerequisites

If you're using a UNIX-style system, you probably already have the `ssh-keygen` utility installed. To determine if it's installed, type `ssh-keygen` on the command line. If it's not installed, you can download OpenSSH for UNIX from http://www.openssh.com/portable.html and install it.

If you're using a Windows operating system you will need PuTTY and the PuTTY Key Generator. Download PuTTY and PuTTYgen from http://www.putty.org and install them.

Creating an SSH Key Pair on the Command Line

1. Open a shell or terminal for entering the commands.
2. At the prompt, enter `ssh-keygen` and provide a name and passphrase when prompted. The keys will be created with the default values: RSA keys of 2048 bits.

Alternatively, you can type a complete `ssh-keygen` command, for example:

```
ssh-keygen -t rsa -N "" -b 2048 -C "<key_name>" -f <path/root_name>
```

The command arguments are shown in the following table:

Argument	Description
`-t rsa`	Use the RSA algorithm.
`-N "<passphrase>"`	A passphrase to protect the use of the key (like a password). If you don't want to set a passphrase, don't enter anything between the quotes. A passphrase is not required. You can specify one as a security measure to protect the private key from unauthorized use.
`-b 2048`	Generate a 2048-bit key. You don't have to set this if 2048 is acceptable, as 2048 is the default. A minimum of 2048 bits is recommended for SSH-2 RSA.
`-C "<key_name>"`	A name to identify the key.
`-f <path/root_name>`	The location where the key pair will be saved and the root name for the files.

Creating an SSH Key Pair Using PuTTY Key Generator

1. Find `puttygen.exe` in the PuTTY folder on your computer, for example, `C:\Program Files (x86)\PuTTY`. Double-click `puttygen.exe` to open it.
2. Specify a key type of SSH-2 RSA and a key size of 2048 bits:
   - In the **Key** menu, confirm that the default value of **SSH-2 RSA key** is selected.
   - For the **Type of key to generate**, accept the default key type of **RSA**.
   - Set the **Number of bits in a generated key** to 2048 if it is not already set.
3. Click **Generate**.
4. Move your mouse around the blank area in the PuTTY window to generate random data in the key.
   When the key is generated, it appears under **Public key for pasting into OpenSSH authorized_keys file**.
5. A **Key comment** is generated for you, including the date and time stamp. You can keep the default comment or replace it with your own more descriptive comment.
6. Leave the **Key passphrase** field blank.
7. Click **Save private key**, and then click **Yes** in the prompt about saving the key without a passphrase.
   The key pair is saved in the PuTTY Private Key (PPK) format, which is a proprietary format that works only with the PuTTY tool set.
   You can name the key anything you want, but use the `ppk` file extension. For example, `mykey.ppk`.
8. Select *all* of the generated key that appears under **Public key for pasting into OpenSSH authorized_keys file**, copy it using **Ctrl + C**, paste it into a text file, and then save the file in the same location as the private key.
   (Do not use **Save public key** because it does not save the key in the OpenSSH format.)
   You can name the key anything you want, but for consistency, use the same name as the private key and a file extension of `pub`. For example, `mykey.pub`.

9. Write down the names and location of your public and private key files. You will need the public key when launching an instance. You will need the private key to access the instance via SSH.

Now that you have a key pair, you're ready to launch instances as described in Creating an Instance.

Creating an Instance

You can create an instance using the Console or API. When you create an instance, it is automatically attached to a virtual network interface card (VNIC) in the cloud network's subnet and given a private IP address from the subnet's CIDR. You can either let the address be automatically assigned, or specify a particular address of your choice. The private IP address lets instances within the cloud network communicate with each other. They can instead use fully qualified domain names (FQDNs) if you've set up the cloud network for DNS (see DNS in Your Virtual Cloud Network).

If the subnet is public, you can optionally assign the instance a public IP address. A public IP address is required to communicate with the instance over the Internet, and to establish a Secure Shell (SSH) or RDP connection to the instance from outside the cloud network. For more information, see Access to the Internet.



Tip

If this is your first time creating an instance, consider following the Getting Started Tutorial for a guided workflow through the steps required to create an instance.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK,

CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.



Tip

When you create an instance, several other resources are involved, such as an image, a cloud network, and a subnet. Those other resources can be in the same compartment with the instance or in other compartments. You must have the required level of access to each of the compartments involved in order to launch the instance. This is also true when you attach a volume to an instance; they don't have to be in the same compartment, but if they're not, you need the required level of access to each of the compartments.

For administrators: The simplest policy to enable users to create instances is listed in Let users launch Compute instances. It gives the specified group general access to managing instances and images, along with the required level of access to attach existing block volumes to the instances. If the group needs to *create* block volumes, they'll need the ability to *manage* block volumes (see Let volume admins manage block volumes, backups, and volume groups).

Partner Image Catalog

If the group needs to create instances based on partner images, they'll need the *manage* permission for app-catalog-listing to create subscriptions to images from the Partner Image catalog. See Let users list and subscribe to images from the Partner Image catalog.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Using the Console

To create a Linux or Windows instance, see the corresponding section that follows:

Creating a Linux Instance

Prerequisites

To create a Linux instance, you'll need:

- A cloud network to launch the instance into. For information about setting up cloud networks, see Overview of Networking.
- The public key, in OpenSSH format, from the key pair that you plan to use for connecting to the instance via SSH. The following sample public key is abbreviated for readability:

  ```
  ssh-rsa AAAAB3NzaC1yc2EAAAABJQAA....lo/gKMLVM2xzc1xJr/Hc26biw3TXWGEakrK1OQ== rsa-key-20160304
  ```

  For information about generating a key pair, see Managing Key Pairs on Linux Instances.

To create a Linux instance using the Console:

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**. Choose a **Compartment** you have permission to work in, and then click **Create Instance**.
2. In the **Create Compute Instance** dialog box, you specify the resources to use for your instance. By default, your instance launches in the current compartment, and the resources you choose also come from the current compartment.
   In the **Create Compute Instance** dialog box, you can specify the following:
   - **Name your instance:** The name for the instance. You can add or change the name later. The name doesn't need to be unique; an Oracle Cloud Identifier (OCID) uniquely identifies the instance.

- **Select an availability domain for your instance:** The availability domain in which you want to run the instance.
- **Choose an operating system or image source:** The source of the image to use for booting the instance. When you click **Change Image Source**, the **Browse All Images** dialog box opens with the operating system or image source options. The following options are available:
  - **Platform Images:** Pre-built images for Oracle Cloud Infrastructure. See Oracle-Provided Images for a list of these images.
  - **Partner Images:** Trusted third-party images published by Oracle partners.
  - **Oracle Images:** Pre-built Oracle enterprise images and solutions enabled for Oracle Cloud Infrastructure.
  - **Custom Images:** Custom images created or imported into your Oracle Cloud Infrastructure environment. See Managing Custom Images for more information.
  - **Boot Volumes:** Boot volumes available for creating a new instance in your Oracle Cloud Infrastructure environment. See Boot Volumes for more information.
  - **Image OCID:** Create an instance using a specific version of an image by providing the image OCID. See Oracle-Provided Image Release Notes to determine the image OCID for Oracle-provided images.
- **Choose instance type:** Select **Virtual Machine** or **Bare Metal Machine**.
- **Choose instance shape:** A template that determines the number of CPUs, amount of memory, and other resources allocated to a newly created instance. When you click **Change Shape**, the **Browse All Shapes** dialog box opens with a list of the virtual machine (VM) or bare metal shapes that are available for the instance type that you selected. Incompatible shapes don't appear in the list. See Compute Shapes for more information about bare metal and VM shapes. Choose a shape and then click **Select Shape**.
- **Configure boot volume:** Size and encryption options for the instance's boot

volume.

To specify a custom size for the boot volume, select the **Custom boot volume size (in GB)** check box. Then, enter a custom size from 50 GB to 32 TB. The specified size must be larger than the default boot volume size for the selected image. See Custom Boot Volume Sizes for more information.

For VM instances, you can optionally select the **Use in-transit encryption** check box. See Block Volume Encryption for more information. If you are using your own Key Management encryption key for the boot volume, then this key is also used for in-transit encryption. Otherwise, the Oracle-provided encryption key is used.

Boot volumes are encrypted by default, but you can optionally encrypt the data in this volume using your own Key Management encryption key. To use Key Management for your encryption needs, select the **Encrypt using Key Management** check box. Then, select the **Vault Compartment** and **Vault** that contain the master encryption key you want to use. Also select the **Master Encryption Key Compartment** and **Master Encryption Key**. For more information about encryption, see Overview of Key Management. If you enable this option, this key is used for both data at rest encryption and in-transit encryption.

- **Add SSH key:** The public key portion of the key pair that you want to use for SSH access to the instance. You can drag and drop single key files into the box. To provide multiple keys, click **Choose files**, then press and hold down the Command key (on Mac) or the CTRL key (on Windows) while selecting files.
- **Configure networking:** The network details for the instance. In this section, you configure the following:
  - **Virtual Cloud Network Compartment:** The compartment containing the network in which to create the instance.
  - **Virtual Cloud Network:** The network in which to create the instance.
  - **Subnet Compartment:** The compartment containing a subnet within the cloud network to attach the instance to.

- **Subnet:** A subnet within the cloud network to attach the instance to. The subnets are either public or private. Private means the instances in that subnet can't have public IP addresses. For more information, see Access to the Internet. Subnets can also be either AD-specific or regional (regional ones have "*regional*" after the name). Oracle recommends using regional subnets. For more information, see About Regional Subnets.

- **Show Advanced Options:** Advanced networking and management options.

  On the **Management** tab, configure the following:

  - **Choose a compartment for your instance:** The compartment in which you want to launch the instance.
  - **Choose a fault domain:** The fault domain to use for the instance. If you do not specify the fault domain, the system selects one for you. After the instance has been created, if you want to change the fault domain you need to terminate the instance and launch a new instance in the preferred fault domain. For more information, see Fault Domains and Best Practices for Your Compute Instance.
  - **User Data:** Data to be used by cloud-init to run custom scripts or provide custom cloud-init configuration. Click **Browse** to select the script file, or paste the script into the text box. The file or script does not need to be base64-encoded, because the Console performs this encoding when the information is submitted. For information about how to take advantage of user data, see the cloud-init documentation.
  - **Enable monitoring**: Select this check box to collect metrics for this instance. When enabled, the OracleCloudAgent software on the instance emits metrics for this instance to the Monitoring service using the oci_computeagent metric namespace.

    This option is available for supported images only. Legacy versions of supported images may also require installation of the OracleCloudAgent software. For more information, see Enabling Monitoring for Compute Instances.

- **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

On the **Networking** tab, configure the following:

- **Private IP Address:** Optional. An available private IP address of your choice from the subnet's CIDR. If you don't specify a value, the private IP address is automatically assigned.
- **Assign public IP address:** Whether to assign the instance a public IP address. Available only if the subnet is public. For more information, see Access to the Internet.
- **Hostname:** Optional. A hostname to be used for DNS within the cloud network. Available only if the VCN and subnet both have DNS labels. For more information, see DNS in Your Virtual Cloud Network.

On the **Image** tab, you can optionally change the image build. By default, the latest build of the image is used to create the instance. You can select an older build of the image that is compatible with the shape you selected. Only compatible image builds are displayed in the list. You need to select a shape before you can change the image build.

3. Click **Create**.

After the instance is provisioned, details about it appear in the instance list. To view additional details, including IP addresses, click the instance name.

When the instance is fully provisioned and running, you can connect to it using SSH as described in Connecting to an Instance.

You also can attach a volume to the instance, provided the volume is in the same availability domain. For background information about volumes, see Overview of Block Volume.

Creating a Windows Instance

Prerequisites

To create a Windows instance, you'll need:

- A cloud network to launch the instance into. For information about setting up cloud networks, see Overview of Networking.
- A security list that enables Remote Desktop Protocol (RDP) access so you can connect to your instance. Specifically, you need a stateful ingress rule for TCP traffic on destination port 3389 from source 0.0.0.0/0 and any source port. For more information, see Security Lists.

To enable RDP access

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the cloud network you're interested in.
3. Under **Resources**, click **Security Lists**. Then click the security list you're interested in.
4. Click **Edit All Rules**.
5. Under **Allow Rules for Ingress**, click **+ Another Ingress Rule**.
6. Enter the following values for the rule:
    a. **Source Type:** CIDR
    b. **Source CIDR:** 0.0.0.0/0
    c. **IP Protocol:** RDP (TCP/3389)
    d. **Source Port Range:** All
    e. **Destination Port Range:** 3389
7. When done, click **Save Security List Rules**.

To create a Windows instance using the Console:

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**. Choose a **Compartment** you have permission to work in, and then click **Create Instance**.
2. In the **Create Compute Instance** dialog box, you specify the resources to use for your instance. By default, your instance launches in the current compartment, and the resources you choose also come from the current compartment.
   In the **Create Compute Instance** dialog box, you can specify the following:
   - **Name your instance:** The name for the instance. You can add or change the name later. The name doesn't need to be unique; an Oracle Cloud Identifier (OCID) uniquely identifies the instance.
   - **Select an availability domain for your instance:** The availability domain in which you want to run the instance.
   - **Choose an operating system or image source:** The source of the image to use for booting the instance. When you click **Change Image Source**, the **Browse All Images** dialog box opens with the operating system or image source options. The following options are available:
     - **Platform Images:** Pre-built images for Oracle Cloud Infrastructure. See Oracle-Provided Images for a list of these images.
     - **Partner Images:** Trusted third-party images published by Oracle partners.
     - **Oracle Images:** Pre-built Oracle enterprise images and solutions enabled for Oracle Cloud Infrastructure.
     - **Custom Images:**Custom images created or imported into your Oracle Cloud Infrastructure environment. See Managing Custom Images for more information.
     - **Boot Volumes:** Boot volumes available for creating a new instance in your Oracle Cloud Infrastructure environment. See Boot Volumes for more information.
     - **Image OCID:** Create an instance using a specific version of an image by

providing the image OCID. See Oracle-Provided Image Release Notes to determine the image OCID for Oracle-provided images.

- **Choose instance type:** Select **Virtual Machine** or **Bare Metal Machine**.
- **Choose instance shape:** A template that determines the number of CPUs, amount of memory, and other resources allocated to a newly created instance.

  When you click **Change Shape**, the **Browse All Shapes** dialog box opens with a list of virtual machine (VM) or bare metal shapes that are available for the instance type that you selected. Incompatible shapes don't appear in the list. See Compute Shapes for more information about bare metal and VM shapes. Choose a shape and then click **Select Shape**.
- **Configure boot volume:** Size and encryption options for the instance's boot volume.

  To specify a custom size for the boot volume, select the **Custom boot volume size (in GB)** check box. Then, enter a custom size from 50 GB (256 GB for Oracle-provided Windows images) to 32 TB. The specified size must be larger than the selected image's default boot volume size. See Custom Boot Volume Sizes for more information.

  For VM instances, you can optionally select the **Use in-transit encryption** check box. See Block Volume Encryption for more information. If you are using your own Key Management encryption key for the boot volume, then this key is also used for in-transit encryption. Otherwise, the Oracle-provided encryption key is used.

  Boot volumes are encrypted by default, but you can optionally encrypt the data in this volume using your own Key Management encryption key. To use Key Management for your encryption needs, select the **Encrypt using Key Management** check box. Then, select the **Vault Compartment** and **Vault** that contain the master encryption key you want to use. Also select the **Master Encryption Key Compartment** and **Master Encryption Key**. For more information about encryption, see Overview of Key Management.
- **Add SSH key:** The public key portion of the key pair that you want to use for SSH access to the instance. You can drag and drop single key files into the box. To

provide multiple keys, click **Choose files**, then press and hold down the Command key (on Mac) or the CTRL key (on Windows) while selecting files.

- **Configure networking:** The network details for the instance. In this section, you configure the following:
  - **Virtual Cloud Network Compartment:** The compartment containing the network in which to create the instance.
  - **Virtual Cloud Network:** The network in which to create the instance.
  - **Subnet Compartment:** The compartment containing a subnet within the cloud network to attach the instance to.
  - **Subnet:** A subnet within the cloud network to attach the instance to. The subnets are either public or private. Private means the instances in that subnet can't have public IP addresses. For more information, see Access to the Internet. Subnets can also be either AD-specific or regional (regional ones have "*regional*" after the name). Oracle recommends using regional subnets. For more information, see About Regional Subnets.
- **Show Advanced Options:** Advanced networking and management options.
  On the **Management** tab, configure the following:
  - **Choose a compartment for your instance:** The compartment in which you want to launch the instance.
  - **Choose a fault domain:** The fault domain to use for the instance. If you do not specify the fault domain, the system selects one for you. After the instance has been created, if you want to change the fault domain you need to terminate the instance and launch a new instance in the preferred fault domain. For more information, see Fault Domains and Best Practices for Your Compute Instance.
  - **User Data:** Data to be used by cloudbase-init to run custom scripts or provide custom cloudbase-init configuration. Click **Browse** to select the script file, or paste the script into the text box. The file or script does not need to be base64-encoded, because the Console performs this encoding

when the information is submitted. For information about how to take advantage of user data, see the cloudbase-init documentation.



Warning

Do not include anything in the script that could trigger a reboot as this could impact the instance launch, causing it to fail. Any actions requiring a reboot should only be performed once the instance state is **RUNNING**.

- **Enable monitoring**: Select this check box to collect metrics for this instance. When enabled, the OracleCloudAgent software on the instance emits metrics for this instance to the Monitoring service using the oci_computeagent metric namespace.

  This option is available for supported images only. Legacy versions of supported images may also require installation of the OracleCloudAgent software. For more information, see Enabling Monitoring for Compute Instances.

- **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

On the **Networking** tab, configure the following:

- **Private IP Address:** Optional. An available private IP address of your choice from the subnet's CIDR. If you don't specify a value, the private IP address is automatically assigned.

- **Assign public IP address:** Whether to assign the instance a public IP address. Available only if the subnet is public. For more information, see Access to the Internet.
- **Hostname:** Optional. A hostname to be used for DNS within the cloud network. Available only if the VCN and subnet both have DNS labels. For more information, see DNS in Your Virtual Cloud Network.

On the **Image** tab, you can optionally change the image build. By default, the latest build of the image is used to create the instance. You can select an older build of the image that is compatible with the shape you selected. Only compatible image builds are displayed in the list. You need to select a shape before you can change the image build.

3. Click **Create**.

After the instance is provisioned, details about it appear in the instance list. To view additional details, including IP addresses and the initial Windows password, click the instance name.

When the instance is fully provisioned and running, you can connect to it using Remote Desktop as described in Connecting to an Instance.

You also can attach a volume to the instance, provided the volume is in the same availability domain. For background information about volumes, see Overview of Block Volume.

Managing Tags for an Instance

You can apply tags to your resources, such as instances, to help you organize them according to your business needs. You can apply tags when you create an instance, or you can update the instance later with the tags that you want.

To manage tags for an instance

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.

2. Click the instance that you're interested in.
3. Click the **Tags** tab to view or edit the existing tags. Or click **Apply tag(s)** to add new ones.

For more information, see Resource Tags.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage instances:

- ListInstances
- LaunchInstance
- GetInstance
- UpdateInstance
- TerminateInstance
- GetWindowsInstanceInitialCredentials

Oracle Cloud Infrastructure enables you to launch instances from images published by Oracle partners from the Partner Image catalog. Use these APIs to work with the Partner Image catalog listings:

- AppCatalogListing
- AppCatalogListingResourceVersion
- AppCatalogListingResourceVersionAgreements
- AppCatalogListingResourceVersionSummary
- AppCatalogListingSummary
- AppCatalogSubscription
- AppCatalogSubscription

Managing Compute Instances

You can simplify the management of your Compute instances using resources such as instance configurations and instance pools.

An instance configuration is a template that defines the settings to use when creating Compute instances.

An instance pool is a group of instances within the same region that are created based off of an instance configuration.

Instance Configurations

An instance configuration defines the settings to use when creating Compute instances as part of an instance pool, including details such as the base image, shape, and metadata. You can also specify the associated resources for the instance, such as block volume attachments and network configuration.

You create an instance configuration from an existing Compute instance. For steps, see Creating an Instance Configuration.

To modify an existing instance configuration, create a new instance configuration with the desired settings.

For steps to delete an instance configuration, see Deleting an Instance Configuration.

Use these API operations to work with instance configurations:

- CreateInstanceConfiguration
- DeleteInstanceConfiguration
- GetInstanceConfiguration
- UpdateInstanceConfiguration
- ListInstanceConfigurations

Instance Pools

Instance pools give you the ability to provision and create multiple Compute instances based off of the same instance configuration, within the same region. They also enable integration with other services, such as the Load Balancing service and IAM service, making it easier to manage groups of instances.

You create an instance pool using an existing instance configuration. For steps, see Creating an Instance Pool.

You can automatically adjust the number of instances in an instance pool based on performance metrics such as CPU utilization. To do this, you enable autoscaling for the instance pool. For background information and steps, see Autoscaling.

After you have created an instance pool, you can update the size and attach or detach load balancers from the Console. To update additional settings, you need to use the CLI, API, or SDKs.

If you need to update the instance configuration, create a new instance configuration and then update the instance pool to use the new instance configuration. For more information, see Updating an Instance Pool.

For steps to delete an instance pool, see Deleting an Instance Pool.



Warning

When you delete an instance pool all of its resources will be permanently deleted, including associated instances, attached boot volumes, and block volumes.

Instance Pool Lifecycle States

The following list describes the different lifecycle states for instance pools.

- **Provisioning:** When you create an instance pool, this is the first state the instance pool is in. Instances for the instance pool are being configured based on the specified instance configuration.
- **Starting:** The instances are being launched. At this point, the only action you can take is to terminate the instance pool.
- **Running:** The instances are created and running.
- **Stopping:** The instances are in the process of being shut down.
- **Stopped:** The instances are shut down.
- **Scaling:** Once an instance pool has been created, if you update the instance pool size, it will go into this state while creating (for increases in size) or terminating (for decreases in size) instances. At this point, the only action you can take is to terminate the instance pool.
- **Terminating:** The instances and associated resources are being terminated.
- **Terminated:** The instance pool, all its instances and associated resources are terminated.

When working with instance configurations and instance pools, keep the following in mind:

- You can't delete an instance configuration if it is associated with at least one instance pool.
- You can use the same instance configuration for multiple instance pools. However, an instance pool can only have one instance configuration associated with it.
- Instance configurations must be in the same compartment where you want to create the associated instance pool.
- If the instance pool has been in the scaling or provisioning state for an extended period of time, it may be because the number of instances requested has exceeded your tenancy's service limits for that shape and availability domain. Check your tenancy's service limits for Compute. If you want to request a limit increase, you need to contact support. For more information, see Service Limits. If this occurs, you need to terminate the instance pool and re-create it.
- If you modify the instance configuration for an instance pool, existing instances that are

part of that pool will not change. Any new instances created after the instance configuration change will use the new instance configuration. New instances will not be created unless you have increased the size of the instance pool or terminate existing instances.

- If you decrease the size of an instance pool, the oldest instances will be terminated first.

Use these API operations to manage instance pools:

- CreateInstancePool
- GetInstancePool
- ResetInstancePool
- SoftresetInstancePool
- StartInstancePool
- StopInstancePool
- TerminateInstancePool
- UpdateInstancePool
- ListInstancePools
- ListInstancePoolInstances
- AttachLoadBalancer
- DetachLoadBalancer

Creating an Instance Configuration

Instance configurations allow you to define the configuration to use when creating Compute instances as part of an instance pool, see Managing Compute Instances for more information.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK,

CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: For a typical policy that gives access to instance pools and instance configurations, see Let users manage Compute instance configurations and instance pools.

Using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
2. In the list of instances, find the instance you want to use to create the instance configuration.
3. Click the highlighted name of the instance to display the instance details.
4. Click **Create Instance Configuration**.
5. Select the compartment you want to create the instance configuration in.
6. Specify a name for the instance configuration. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
7. Click **Create Instance Configuration**.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the CreateInstanceConfiguration operation to create an instance configuration.

Creating an Instance Pool

Instance pools give you the ability to provision and create multiple Compute instances based off of the same configuration, within the same region. See Managing Compute Instances for more information.

Optionally, you can associate a load balancer with an instance pool. If you do this, when you add an instance to the instance pool, the instance is automatically added to the load balancer's backend set . After the instance reaches a healthy state (the instance is listening on the configured port number), incoming traffic is automatically routed to the new instance. For background information about the Load Balancing service, see Overview of Load Balancing.



Note

There are some values for instance configuration settings that are incompatible with instance pools. If you try to create an instance pool with one or more of these settings, an error will occur. See Error occurs when creating an instance pool with a valid instance configuration for more information.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: For a typical policy that gives access to instance pools and instance configurations, see Let users manage Compute instance configurations and instance pools.

Prerequisites

Before you can create an instance pool, you need:

- An instance configuration. The instance configuration must be in the same compartment where you want to create the instance pool. For more information, see Creating an Instance Configuration.

- If you want to associate the instance pool with a load balancer, you need a load balancer and backend set. For steps to create a load balancer, see Managing a Load Balancer.

Using the Console

You can create an instance pool from the **Instance Configuration Details** page or from the **Instance Pools** page.

To create an instance pool from the Instance Configuration Details page

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instance Configurations**.
2. Click the instance configuration that you want to use to create the instance pool.
3. Click **Create Instance Pool**.
4. In the **Create in Compartment** list, select the compartment that you want to create the instance configuration in. This must be the same compartment as the instance configuration you're using.
5. Specify a name for the instance pool. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API).
6. Specify the targeted number of instances for the instance pool.

7. If you want to associate a load balancer with the instance pool, select **Attach a Load Balancer**. Then, enter the following:
   - **Load Balancer Compartment:** The compartment that the load balancer is in.
   - **Load Balancer:** The load balancer to associate with the instance pool.
   - **Backend Set:** The name of the backend set on the load balancer to add instances to.
   - **Port:** The server port on the instances to which the load balancer must direct traffic. This value applies to all instances that use this load balancer attachment.
   - **VNIC:** The VNIC to use when adding the instance to the backend set. Instances that belong to a backend set are also called backend servers. The private IP address is used. This value applies to all instances that use this load balancer attachment.

   For background information about load balancers, see Overview of Load Balancing.
8. In the **Availability Domain** section, enter the following:
   - **Availability Domain:** The availability domain to launch the instances in.
   - **Virtual Cloud Network Compartment:** The compartment that the virtual cloud network (VCN) is in.
   - **Primary VNIC:** The primary **Virtual Cloud Network** and **Subnet** for the instance pool.
   - **Secondary VNIC:** This option appears only if secondary VNICs are defined by the instance configuration. Select the secondary **Virtual Cloud Network** and **Subnet** for the instance pool.

   If you want to configure the instance pool to create instances in more than one availability domain, click **+ Additional Selection** and select a different availability domain for the instance pool. Then, specify the VCN details for the second availability domain.
9. Click **Create Instance Pool**.

To create an instance pool from the Instance Pools page

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instance Pools**.
2. Click **Create Instance Pool**.
3. In the **Create in Compartment** list, select the compartment that you want to create the instance pool in. This must be the same compartment as the instance configuration you're using.
4. Specify a name for the instance pool. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API).
5. Specify the targeted **Number of Instances** for the instance pool.
6. In the **Instance Configuration Compartment** list, select the compartment that contains the instance configuration that you want to use for creating the instance pool.
7. Select the **Instance Configuration** to use when creating the instance pool's instances.
8. If you want to associate a load balancer with the instance pool, select **Attach a Load Balancer**. Then, enter the following:
   - **Load Balancer Compartment:** The compartment that the load balancer is in.
   - **Load Balancer:** The load balancer to associate with the instance pool.
   - **Backend Set:** The name of the backend set on the load balancer to add instances to.
   - **Port:** The server port on the instances to which the load balancer must direct traffic. This value applies to all instances that use this load balancer attachment.
   - **VNIC:** The VNIC to use when adding the instance to the backend set. Instances that belong to a backend set are also called backend servers. The private IP address is used. This value applies to all instances that use this load balancer attachment.

   For background information about load balancers, see Overview of Load Balancing.

9. In the **Availability Domain** section, enter the following:
    - **Availability Domain:** The availability domain to launch the instances in.
    - **Virtual Cloud Network Compartment:** The compartment that the virtual cloud network (VCN) is in.
    - **Primary VNIC:** The primary **Virtual Cloud Network** and **Subnet** for the instance pool.
    - **Secondary VNIC:** This option appears only if secondary VNICs are defined by the instance configuration. Select the secondary **Virtual Cloud Network** and **Subnet** for the instance pool.

    If you want to configure the instance pool to create instances in more than one availability domain, click **+ Additional Selection** and select a different availability domain for the instance pool. Then, specify the VCN details for the second availability domain.
10. Click **Create Instance Pool**.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to create and manage instance pools:

- CreateInstancePool
- AttachLoadBalancer
- DetachLoadBalancer

Updating an Instance Pool

You can update the number of instances for an instance pool using the Console.

Optionally, you can associate a load balancer with an instance pool. If you do this, when you add an instance to the instance pool, the instance is automatically added to the load balancer's

backend set. After the instance reaches a healthy state (the instance is listening on the configured port number), incoming traffic is automatically routed to the new instance. For background information about the Load Balancing service, see Overview of Load Balancing.

To update other settings for an instance pool, use the command line interface (CLI), SDKs, or REST APIs.

You can automatically adjust the number of instances in an instance pool based on performance metrics such as CPU utilization. To do this, you enable autoscaling for the instance pool. For background information and steps, see Autoscaling.

See Managing Compute Instances for more information about instance pools and instance configurations.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: For a typical policy that gives access to instance pools and instance configurations, see Let users manage Compute instance configurations and instance pools.

Using the Console

To update the instance pool size

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instance Pools**.
2. Click the name of the instance pool that you're interested in.
3. Click **Edit**.
4. Specify the updated number of instances for the instance pool, and then click **Save**

Changes.

When you update the instance pool size, it triggers a scaling event. Keep the following in mind:

- If the instance pool lifecycle state is **RUNNING**, the instance pool will create new instances or terminate existing instances at that time, to match the new size of the instance pool. Instances are terminated in the order that they were created, first-in, first-out.
- If the instance pool lifecycle state is **STOPPED**, for an increase in size, new instances will be configured for the instance pool, but won't be launched. For a decrease in size, the instances will be terminated.



Important

If the instance pool has been in the scaling or provisioning state for an extended period of time, it may be because the number of instances requested has exceeded your tenancy's service limits for that shape and availability domain. Check your tenancy's service limits for Compute.

To attach a load balancer to an instance pool

You must have a load balancer and backend set to associate with the instance pool. For steps to create a load balancer, see Managing a Load Balancer.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instance Pools**.
2. Click the name of the instance pool that you're interested in.
3. In the **Resources** section, click **Load Balancers**.

4. Click **Attach a Load Balancer**.
5. Enter the following:
    - **Load Balancer Compartment:** The compartment that the load balancer is in.
    - **Load Balancer:** The load balancer to associate with the instance pool.
    - **Backend Set:** The name of the backend set on the load balancer to add instances to.
    - **Port:** The server port on the instances to which the load balancer must direct traffic. This value applies to all instances that use this load balancer attachment.
    - **VNIC:** The VNIC to use when adding the instance to the backend set. Instances that belong to a backend set are also called backend servers. The private IP address is used. This value applies to all instances that use this load balancer attachment.
6. Click **Attach**.

To detach a load balancer from an instance pool

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instance Pools**.
2. Click the name of the instance pool that you're interested in.
3. In the **Resources** section, click **Load Balancers**.
4. Click the Actions icon (three dots) for the load balancer.
5. Click **Detach**, and then click **Detach** to confirm.

Using the API

To update other instance pool configuration settings, use the CLI, SDKs, or REST APIs. For information about using the CLI, see Command Line Interface (CLI). For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.



Note

There are some values for instance configuration settings that are incompatible with instance pools. If you try to update an instance pool with one or more of these settings, an error will occur. See Error occurs when creating or updating an instance pool with a valid instance configuration for more information.

For instance pool configuration settings, such as the instance configuration, display name, tags, or availability domain selections, use the UpdateInstancePool operation.

To manage the load balancers that are associated with an instance pool, use the AttachLoadBalancer and DetachLoadBalancer operations.

To update the configuration used by the instance pool when creating instances you can either:

- Create a new instance configuration with the desired settings. You can do this using the Console. For steps, see Creating an Instance Configuration. To do this using the API, use the CreateInstanceConfiguration operation.
- Update the existing instance configuration for the instance pool. You can only update the display name and tags of existing instance configurations. For any other updates, create a new instance configuration with the settings you want to use. To update the display name or tags, use the UpdateInstanceConfiguration operation. You cannot use the Console to update instance configuration settings.

Deleting an Instance Configuration

You can permanently delete instance configurations that you no longer need.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK,

CLI, or other tool. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: For a typical policy that gives access to instance pools and instance configurations, see Let users manage Compute instance configurations and instance pools.

Using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instance Configurations**.
2. In the list of instance configurations, find the instance configuration you want to delete.
3. Click the highlighted name of the instance configuration to display the instance configuration details.
4. Click **Delete**, and then click **Delete Instance Configuration**.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the DeleteInstanceConfiguration operation to delete an instance configuration.

Deleting an Instance Pool

You can permanently delete instance pools that you no longer need.



Warning

When you delete an instance pool all of its resources will be permanently deleted, including associated instances, attached boot volumes, and block volumes.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: For a typical policy that gives access to instance pools and instance configurations, see Let users manage Compute instance configurations and instance pools.

Using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instance Pools**.
2. Click the instance pool that you want to delete.
3. Click **Actions**, and then click **Terminate**.
4. Enter the name of the instance pool and click **Terminate**.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the DeleteInstanceConfiguration operation to delete an instance configuration.

Autoscaling

Autoscaling enables you to automatically adjust the number of Compute instances in an instance pool based on performance metrics such as CPU utilization. This helps you provide consistent performance for your end users during periods of high demand, and helps you reduce your costs during periods of low demand.

You select a performance metric to monitor, and set thresholds that the performance metric must reach to trigger an autoscaling event. When system usage meets a threshold,

autoscaling dynamically allocates resources in near-real time. As load increases, instances are automatically provisioned: the instance pool *scales out*. As load decreases, instances are automatically removed: the instance pool *scales in*.

Autoscaling relies on performance metrics that are collected by the Monitoring service. These performance metrics are aggregated into one-minute time periods and then averaged across the instance pool. When three consecutive values (that is, the average metrics for three consecutive minutes) meet the threshold, an autoscaling event is triggered.

A cooldown period between autoscaling events lets the system stabilize at the updated level. The cooldown period starts when the instance pool reaches a steady state. Autoscaling continues to evaluate performance metrics during the cooldown period. When the cooldown period ends, autoscaling adjusts the instance pool's size again if needed.

For background information about the Monitoring service, see Monitoring Overview. For information about the metrics emitted by Compute instances, see Compute Metrics.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: For a typical policy that gives access to autoscaling configurations, see Let users manage Compute autoscaling configurations.

Tagging Resources

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

Prerequisites

- You have an instance pool. Optionally, you can attach a load balancer to the instance pool. For steps to create an instance pool and attach a load balancer, see Creating an Instance Pool.
- Monitoring is enabled on the instances in the instance pool. For steps to enable monitoring, see Enabling Monitoring for Compute Instances.
- The instance pool supports the maximum number of instances that you want to scale to. This limit is determined by your tenancy's service limits. For more information, see Service Limits.

Using the Console

To create an autoscaling configuration

Each instance pool can have one autoscaling configuration.

You can create an autoscaling configuration from the **Instance Pools** page or the **Autoscaling Configurations** page.

To create an autoscaling configuration from the Instance Pools page

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instance Pools**.
2. Click the instance pool that you want to apply an autoscaling configuration to.

3. Click **Actions**, and then click **Create Autoscaling Configuration**.
4. Select the **Compartment** that you want to create the autoscaling configuration in. This must be the same compartment as the instance pool you're scaling.
5. Enter a name for the autoscaling configuration.
6. Select the **Instance Pool** to apply the autoscaling configuration to. The current instance pool is selected by default.
7. In the **Cooldown in Seconds** box, enter the minimum amount of time to wait between scaling events. The cooldown period gives the system time to stabilize before rescaling. The minimum value is 300 seconds.
8. In the **Autoscaling Policy** area, define the criteria that trigger autoscaling actions and the actions to take:
    a. Enter a name for the autoscaling policy.
    b. Select the **Performance Metric** that triggers an increase or decrease in the number of instances in the instance pool. For example, CPU utilization or memory utilization.
    c. In the **Scaling Limits** area, specify the number of instances in the instance pool:
        - **Minimum Number of Instances:** The minimum number of instances that the instance pool is allowed to decrease to.
        - **Maximum Number of Instances:** The maximum number of instances that the instance pool is allowed to increase to. The maximum number of instances depends on the limits for your tenancy. For more information, see Service Limits.
        - **Initial Number of Instances:** The number of instances to launch in the instance pool immediately after autoscaling is enabled. After autoscaling retrieves performance metrics, the number of instances is automatically adjusted from this initial number to a number that is based on the limits that you set.

d. In the **Scaling Rule** area, specify the thresholds that the performance metric must reach to trigger a scaling event:
   - Select a **Scale-Out Operator** and **Threshold Percentage** at which to increase the number of instances. Then enter the **Number of Instances to Add**. For example, when CPU utilization is greater than 90%, add 10 instances to the instance pool.
   - Select a **Scale-In Operator** and **Threshold Percentage** at which to decrease the number of instances. Then enter the **Number of Instances to Remove**. For example, when CPU utilization is less than 50%, remove 5 instances from the instance pool.

9. **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
10. Click **Create**.
    Autoscaling runs. The cooldown period starts when the instance pool's status changes from **SCALING** to **RUNNING**.

To create an autoscaling configuration from the Autoscaling Configuration page

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Autoscaling Configurations**.
2. Click **Create Autoscaling Configuration**.
3. Select the **Compartment** that you want to create the autoscaling configuration in. This must be the same compartment as the instance pool you're scaling.
4. Enter a name for the autoscaling configuration.
5. Select the **Instance Pool** to apply the autoscaling configuration to.

6. In the **Cooldown in Seconds** box, enter the minimum amount of time to wait between scaling events. The cooldown period gives the system time to stabilize before rescaling. The minimum value is 300 seconds.
7. In the **Autoscaling Policy** area, define the criteria that trigger autoscaling actions and the actions to take:
    a. Enter a name for the autoscaling policy.
    b. Select the **Performance Metric** that triggers an increase or decrease in the number of instances in the instance pool. For example, CPU utilization or memory utilization.
    c. In the **Scaling Limits** area, specify the number of instances in the instance pool:
        - **Minimum Number of Instances:** The minimum number of instances that the instance pool is allowed to decrease to.
        - **Maximum Number of Instances:** The maximum number of instances that the instance pool is allowed to increase to. The maximum number of instances depends on the limits for your tenancy. For more information, see Service Limits.
        - **Initial Number of Instances:** The number of instances to launch in the instance pool immediately after autoscaling is enabled. After autoscaling retrieves performance metrics, the number of instances is automatically adjusted from this initial number to a number that is based on the limits that you set.

d. In the **Scaling Rule** area, specify the thresholds that the performance metric must reach to trigger a scaling event:
   - Select a **Scale-Out Operator** and **Threshold Percentage** at which to increase the number of instances. Then enter the **Number of Instances to Add**. For example, when CPU utilization is greater than 90%, add 10 instances to the instance pool.
   - Select a **Scale-In Operator** and **Threshold Percentage** at which to decrease the number of instances. Then enter the **Number of Instances to Remove**. For example, when CPU utilization is less than 50%, remove 5 instances from the instance pool.

8. In the **Autoscaling Policy** area, define the criteria that trigger autoscaling actions and the actions to take:
   a. Enter a name for the autoscaling policy.
   b. Select the **Performance Metric** that triggers an increase or decrease in the number of instances in the instance pool. For example, CPU utilization or memory utilization.
   c. In the **Scaling Limits** area, specify the number of instances in the instance pool:
      - **Minimum Number of Instances:** The minimum number of instances that the instance pool is allowed to decrease to.
      - **Maximum Number of Instances:** The maximum number of instances that the instance pool is allowed to increase to. The maximum number of instances depends on the limits for your tenancy. For more information, see Service Limits.
      - **Initial Number of Instances:** The number of instances to launch in the instance pool immediately after autoscaling is enabled. After autoscaling retrieves performance metrics, the number of instances is automatically adjusted from this initial number to a number that is based on the limits that you set.

d. In the **Scaling Rule** area, specify the thresholds that the performance metric must reach to trigger a scaling event:
   - Select a **Scale-Out Operator** and **Threshold Percentage** at which to increase the number of instances. Then enter the **Number of Instances to Add**. For example, when CPU utilization is greater than 90%, add 10 instances to the instance pool.
   - Select a **Scale-In Operator** and **Threshold Percentage** at which to decrease the number of instances. Then enter the **Number of Instances to Remove**. For example, when CPU utilization is less than 50%, remove 5 instances from the instance pool.

9. **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
10. Click **Create**.
    Autoscaling runs. The cooldown period starts when the instance pool's status changes from **SCALING** to **RUNNING**.

To edit an autoscaling configuration

You can change these characteristics of an autoscaling configuration:

- Name
- Cooldown period between autoscaling actions

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Autoscaling Configurations**.
2. Click the autoscaling configuration that you're interested in.

3. Click **Edit**.
4. Update the **Name** or **Cooldown in Seconds**, and then click **Save Changes**.

To edit an autoscaling policy

You can change these characteristics of an autoscaling policy:

- Name
- Which performance metric triggers an autoscaling action
- Scale-out and scale-in operators and thresholds
- The number of instances to add or remove

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Autoscaling Configurations**.
2. Click the autoscaling configuration that you're interested in.
3. In the **Autoscaling Policies** area, click **Edit**.
4. Make your changes, and then click **Save**.

To disable an autoscaling configuration

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Autoscaling Configurations**.
2. Click the autoscaling configuration that you're interested in.
3. Click **Disable**, and then confirm when prompted.

To delete an autoscaling configuration

When you delete an autoscaling configuration, the instance pool remains in its most recent state.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Autoscaling Configurations**.
2. Click the autoscaling configuration that you're interested in.
3. Click **Delete**, and then confirm when prompted.

To manage tags for an autoscaling configuration

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Autoscaling Configurations**.
2. Click the autoscaling configuration that you're interested in.
3. Click the **Tags** tab to view or edit the existing tags. Or click **Apply tag(s)** to add new ones.

For more information, see Resource Tags.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the Autoscaling API to manage autoscaling configurations and policies.

Connecting to an Instance

You can connect to a running instance by using a Secure Shell (SSH) or Remote Desktop connection. Most UNIX-style systems include an SSH client by default. To connect to a Linux instance from a Windows system, you can download a free SSH client called PuTTY from http://www.putty.org.

Required IAM Policy

To connect to a running instance with SSH, you don't need an IAM policy to grant you access. However, to SSH you need the public IP address of the instance (see Prerequisites below). If there's a policy that lets you launch an instance, that policy probably also lets you get the instance's IP address. The simplest policy that does both is listed in Let users launch Compute instances.

For administrators: Here's a more restrictive policy that lets the specified group get the IP address of existing instances and use power actions on the instances (e.g., stop, start, etc.), but not launch or terminate instances. The policy assumes the instances and the cloud network are together in a single compartment (XYZ):

```
Allow group InstanceUsers to read virtual-network-family in compartment XYZ

Allow group InstanceUsers to use instance-family in compartment XYZ
```

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Prerequisites

You'll need the following information to connect to the instance:

- For Linux Instances: The full path to the key pair that you used when you launched the instance. For information about generating key pairs, see Managing Key Pairs on Linux Instances.
- The default user name for the instance. If you used an Oracle-provided Linux, CentOS or Windows image to launch the instance, the user name is `opc`. If you used the Ubuntu image to launch the instance, the user name is `ubuntu`.
- The public IP address of the instance. You can get the address from the list of instances in the Console. Click **Compute**, choose your **Compartment**, and then find your instance in the list. Alternatively, you can use the Core Services API `ListVnicAttachments` and `GetVnic` operations.

Connecting to a Linux Instance

Log in to your instance using SSH.

Connecting to Your Linux Instance from a Unix-style System

1. Use the following command to set the file permissions so that only you can read the file:

   ```
   $ chmod 400 <private_key>
   ```

   <private_key> is the full path and name of the file that contains the private key associated with the instance you want to access.

2. Use the following SSH command to access the instance.

   ```
   $ ssh -i <private_key> <username>@<public-ip-address>
   ```

   <private_key> is the full path and name of the file that contains the private key associated with the instance you want to access.
   <username> is the default name for the instance. For Oracle Linux and CentOS images, the default user name is `opc`. For the Ubuntu image, the default name is `ubuntu`.
   <public-ip-address> is your instance IP address that you retrieved from the Console.

Connecting to Your Linux Instance from a Windows System

1. Open `putty.exe`.
2. In the **Category** pane, select **Window**, and then select **Translation**.
3. In the **Remote character set** drop-down list, select **UTF-8**. The default locale setting on Linux-based instances is UTF-8, and this configures PuTTY to use the same locale.
4. In the **Category** pane, select **Session** and enter the following:
   - **Host Name (or IP address)**:
     <username>@*<public-ip-address>*

<username> is the default name for the instance. For Oracle Linux and CentOS images, the default user name is `opc`. For the Ubuntu image, the default name is `ubuntu`.
<public-ip-address> is your instance public IP address that you retrieved from the Console

- **Port:** 22
- **Connection type:** SSH

5. In the **Category** pane, expand **Connection**, expand **SSH**, and then click **Auth**.
6. Click **Browse**, and then select your private key.
7. Click **Open** to start the session.
   If this is your first time connecting to the instance, you might see a message that the server's host key is not cached in the registry. Click **Yes** to continue the connection.

Connecting to a Windows Instance

You can connect to a Windows instance by using a Remote Desktop connection. Most Windows systems include a Remote Desktop client by default.

To enable Remote Desktop Protocol (RDP) access to the Windows instance, you need to add a stateful ingress rule for TCP traffic on destination port 3389 from source 0.0.0.0/0 and any source port.

To enable RDP access

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the cloud network you're interested in.
3. Under **Resources**, click **Security Lists**. Then click the security list you're interested in.

4. Click **Edit All Rules**.
5. Under **Allow Rules for Ingress**, click **+ Another Ingress Rule**.
6. Enter the following values for the rule:
    a. **Source Type:** CIDR
    b. **Source CIDR:** 0.0.0.0/0
    c. **IP Protocol:** RDP (TCP/3389)
    d. **Source Port Range:** All
    e. **Destination Port Range:** 3389
7. When done, click **Save Security List Rules**.

Connecting to Your Windows Instance from a Remote Desktop Client

1. Open the Remote Desktop client.
2. In the **Computer** field, enter the public IP address of the instance you want to connect to. Your public IP is the instance address you get from the Console.
3. The **User name** is **opc**. Depending on the Remote Desktop client you are using, you might have to connect to the instance before you can enter this credential.
4. Click **Connect** to start the session.
5. Accept the certificate if you are prompted to do so.
6. If you are connecting to the instance for the first time, enter the initial password that was provided to you by Oracle Cloud Infrastructurewhen you launched the instance. You will be prompted to change the password as soon as you log in. Your new password must be at least 12 characters long and must comply with Microsoft's password policy. Otherwise, enter the password you created. If you are using a custom image, you may need to know the password for the instance the image was created from. For details about Windows custom images, see Creating Windows Custom Images.
7. Press **Enter**.

Instance Console Connections

The Oracle Cloud Infrastructure Compute service provides console connections that enable you to remotely troubleshoot malfunctioning instances, such as:

- An imported or customized image that does not complete a successful boot.
- A previously working instance that stops responding.

There are two types of instance console connections:

- Serial console connections
- VNC console connections

Creating the Instance Console Connection

Before you can connect to the serial console or VNC console, you need to create the instance console connection.

To create the console connection for an instance

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
2. In the list of instances, find the instance you want to access the serial console for, and then click the instance name.
3. In the **Resources** section on the **Instance Details** page, click **Console Connections**, and then click **Create Console Connection**.
4. Specify the public key portion for the SSH key, either by browsing and selecting a public key file, for example id_rsa.pub, or by pasting your public key into the text box, and then click **Create Console Connection**.

Once the console connection has been created and is available, the status changes to **ACTIVE**.

Connecting to the Serial Console

Once you have created the console connection for the instance, you can then connect to the serial console by using a Secure Shell (SSH) connection. Once you are finished with the serial console and have terminated the SSH connection, you should delete the serial console connection. If you do not disconnect from the session, Oracle Cloud Infrastructure will terminate the serial console session after 24 hours and you will need to re-authenticate to connect again.



Note

Serial console connections for VM instances launched prior to September 2017

Serial console connections only work for VM instances launched in September 2017 or later.



Note

Serial console connections for bare metal instances launched prior to November 2017

Serial console connections only work for Bare Metal instances launched in November 2017 or later.

Connecting from Mac OS X and Linux Operating Systems

You connect to the serial console by using an SSH client. Mac OS X and most Linux distributions by default include the SSH client OpenSSH.

To connect to the serial console for an instance using OpenSSH on Mac OS X or Linux

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**. Click the instance you want to connect to.
2. On the **Instances Details** page, in the **Resources** section, click **Console Connections**.
3. Click the Actions icon (three dots), and then click **Connect with SSH**.
4. Select **LINUX/MAC OS** for **PLATFORM**.
5. Click **Copy** to copy the string to the clipboard.
6. Paste the connection string copied from the previous step to a terminal window on a Mac OS X or Linux system, and hit enter to connect to the console.

   If you are not using the default SSH key or ssh-agent, you can modify the serial console connection string to include the identity file flag, `-i` to specify the SSH key to use. You need to specify this for both the SSH connection and the SSH ProxyCommand, as shown in the following line:

   ```
   ssh -i /<path>/<ssh_key> -o ProxyCommand='ssh -i /<path>/<ssh_key> -W %h:%p -p 443...
   ```

7. Hit enter again to activate the console.

Connecting from Windows Operating Systems

Windows does not include an SSH client by default, so you need to install one. You can use PuTTY, or there are options that include a version of OpenSSH such as:

- git tools for Windows
- Windows Subsystem for Linux

Note that the steps to connect to the serial console from the PuTTY client will be different than the steps for OpenSSH.

To connect to the serial console for an instance on Microsoft Windows

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**. Click the instance you want to connect to.
2. On the **Instances Details** page, in the **Resources** section, click **Console Connections**.
3. Click the Actions icon (three dots), and then click **Connect with SSH**.
4. If you are using PuTTY, select **WINDOWS** for **PLATFORM**.
   If you are using OpenSSH select **LINUX/MAC OS** for **PLATFORM**.
5. Click **Copy** to copy the string to the clipboard.
6. Paste the connection string copied from the previous step to PuTTY or your OpenSSH client and hit enter to connect to the console.
7. Hit enter again to activate the console.

Connecting to the VNC Console



Warning

The VNC console connection uses SSH port forwarding to create a secure connection from your local system to the VNC server attached to your instance's console. While this is a secure way to use VNC over the internet, owners of multiuser systems should be aware that opening a port on the local system makes it available to all of the users on that system until a VNC client connects. For this reason, we don't recommend using this product on a multiuser system unless you take proper actions to secure the port or you isolate the VNC client by running it in a virtual environment, such as Oracle VM VirtualBox.

Once you have created the console connection for the instance, you need to set up a secure tunnel to the VNC server on the instance, and then you can connect with a VNC client.



Note

VNC console connections for VM instances launched prior to October 13th, 2017

VNC console connections only work for VM instances launched on October 13th, 2017 or later.



Note

VNC console connections for bare metal instances launched prior to February 21, 2019

VNC console connections only work for bare metal instances launched on February 21st, 2019 or later, using one of the following shapes:

- BM.GPU2.2
- BM.HPC2.36
- BM.Standard2.52
- BM.DenseIO2.52

To set up a secure tunnel to the VNC server on the instance using OpenSSH on Mac OS X or Linux

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**. Click the instance you want to connect to.
2. On the **Instances Details** page, in the **Resources** section, click **Console**

Connections.

3. Click the Actions icon (three dots), and then click **Connect with VNC**.
4. Select **LINUX/MAC OS** for **PLATFORM**.
5. Click **Copy** to copy the string to the clipboard.
6. Paste the connection string copied from the previous step to a terminal window on a Mac OS X or Linux system, and hit enter to set up the secure connection.
7. Once the connection has been established, open your VNC client and specify `localhost` as the host to connect to and `5900` as the port to use.



Note

Mac OS X Screen Sharing.app Not Compatible with VNC Console Connections

The Mac OS X built-in VNC client, Screen Sharing.app does not work with VNC console connections in Oracle Cloud Infrastructure. Use another VNC client, such as Real VNC Viewer or Chicken.

To set up a secure tunnel to the VNC server on the instance using PowerShell on Windows

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**. Click the instance you want to connect to.
2. On the **Instances Details** page, in the **Resources** section, click **Console Connections**.
3. Click the Actions icon (three dots), and then click **Connect with VNC**.
4. Select **WINDOWS** for **PLATFORM**.
5. Click **Copy** to copy the string to the clipboard.

6. Paste the connection string copied from the previous step to Windows Powershell and hit enter to set up the secure connection.
7. Once the connection has been established, open your VNC client and specify `localhost` as the host to connect to and `5900` as the port to use.



Note

Secure Connection Warning

When you connect, you may see a warning from the VNC client that the connection is not encrypted. Since you are connecting through SSH, the connection is secure, so this is not an issue.

Troubleshooting Instances from Instance Console Connections

Once you are connected with an instance console connection, you can perform various tasks, such as:

- Edit system configuration files.
- Add or reset the SSH keys for the **opc** user.

Both of these tasks require you to boot into a bash shell, in maintenance mode.



Note

The following tasks describe steps specific to instances running Oracle Linux 7, connecting from OpenSSH. Other OS versions and SSH clients may require different steps.

To boot into maintenance mode

1. Reboot the instance from the Console
    - In the Console, on the **Instances Details page**, click **Reboot**.
2. Once the reboot process starts, switch back to the terminal window, and you see Console messages start to appear in the window. As soon as you see the GRUB boot menu appear, use the up/down arrow key to stop the automatic boot process, enabling you to use the boot menu.
3. In the boot menu, highlight the top item in the menu, and type `e` to edit the boot entry.
4. In edit mode, use the down arrow key to scroll down through the entries until you reach the line that starts with either `linuxefi` for instances running Oracle Linux 7.x, or `kernel` for instances running Oracle Linux 6.x.
5. At the end of that line, add the following:

    ```
    init=/bin/bash
    ```

6. Reboot the instance from the terminal window by entering the keyboard shortcut **CTRL+x**.

Once the instance has rebooted, you'll see the Bash shell command-line prompt, and you can proceed with either of the following procedures.

To edit the system configuration files

1. From the Bash shell, run the following command to load the SELinux policies to preserve the context of the files you are modifying:

    ```
    /usr/sbin/load_policy -i
    ```

2. Run the following command to remount the root partition with read/write permissions:

    ```
    /bin/mount -o remount, rw /
    ```

3. Edit the configuration files as needed to try to recover the instance.
4. Once you have finished editing the configuration files, to start the instance from the existing shell, run the following command:

```
exec /usr/lib/systemd/systemd
```

Alternatively, to reboot the instance, run the following command:

```
/usr/sbin/reboot -f
```

To add or reset the SSH key for the opc user

1. From the Bash shell, run the following command to load the SELinux policies to preserve the context of the files you are modifying:

   ```
   /usr/sbin/load_policy -i
   ```

2. Run the following command to remount the root partition with read/write permissions:

   ```
   /bin/mount -o remount, rw /
   ```

3. From the Bash shell, run the following command to change to the SSH key directory for the **opc** user:

   ```
   cd ~opc/.ssh
   ```

4. Rename the existing authorized keys file with the following command:

   ```
   mv authorized_keys authorized_keys.old
   ```

5. Replace the contents of the public key file with the new public key file with the following command:

   ```
   echo '<contents of .pub key file>' >> authorized_keys
   ```

6. Restart the instance by running the following command:

   ```
   /usr/sbin/reboot -f
   ```

Exiting the Instance Console Connection

To exit the serial console connection

When using SSH, the ~ character at the beginning of a new line is used as an escape

character.

- To exit the serial console, enter:

  ```
  ~.
  ```

- To suspend the SSH session, enter:

  ```
  ~^z
  ```

  The ^ character represents the **CTRL** key

- To see all the SSH escape commands, enter:

  ```
  ~?
  ```

To exit the VNC console connection

1. Close the VNC client.
2. In the Terminal or Powershell window, type `CTRL C`

Once you are finished using the console connection, delete the connection for the instance.

To delete the console connection for an instance

1. In the Console, on the **Instances Details** page, in the **Resources** section, click **Console Connections**.
2. Click the Actions icon (three dots), click **Delete**, and then click **OK** to confirm.

Adding Users on an Instance

If you created your instance using an Oracle-provided Linux or CentOS image, you can use SSH to access your instance from a remote host as the `opc` user. If you created your instance using the Ubuntu image, you can use SSH to access your instance from a remote host as the `ubuntu` user. After logging in, you can add users on your instance.

If you created your instance using an Oracle-provided Windows image, you can create new users after you log on to the instance through a Remote Desktop client.

Creating Additional SSH-Enabled Users on Linux Instances

If you do not want to share your SSH key, you can create additional SSH-enabled users:

- Generate SSH key pairs for the users offline.
- Add the new users.
- Append a public key to the `~/.ssh/authorized_keys` file for each new user.



Tip

If you re-create an instance from an Oracle-provided image, users and SSH public keys that you added or edited manually (that is, users that weren't defined in the machine image) must be added again.

If you need to edit the `~/.ssh/authorized_keys` file of a user on your instance, start a second SSH session before you make any changes to the file and ensure that it remains connected while you edit the file. If the `~/.ssh/authorized_keys` file becomes corrupted or you inadvertently make changes that lock you out of the instance, you can use the backup SSH session to fix or revert the changes. Before closing the backup SSH session, test all changes you made by logging in with the new or updated SSH key.

The new users then can SSH to the instance using the appropriate private keys.

To create an additional SSH-enabled user:

1. Generate an SSH key pair for the new user. See Managing Key Pairs on Linux Instances.
2. Copy the public key value to a text file for use later in this procedure.

3. Log in to your instance. See Connecting to an Instance.
4. Become the root user:

```
sudo su
```

5. Create the new user:

```
useradd <new_user>
```

6. Create a `.ssh` directory in the new user's home directory:

```
mkdir /home/<new_user>/.ssh
```

7. Copy the SSH public key that you saved to a text file into the `/home/new_user/.ssh/authorized_keys` file:

```
echo <public_key> > /home/<new_user>/.ssh/authorized_keys
```

8. Change the owner and group of the `/home/username/.ssh` directory to the new user:

```
chown -R <new_user>:<group> /home/<new_user>/.ssh
```

9. To enable `sudo` privileges for the new user, run the `visudo` command and edit the `/etc/sudoers` file as follows:
    a. In `/etc/sudoers`, look for:

```
%<username> ALL=(ALL) NOPASSWD: ALL
```

    b. Add the following line immediately after the preceding line:

```
%<group> ALL=(ALL) NOPASSWD: ALL
```

You can now log in as the new user.

Creating Additional Users on a Windows Instance

To create a new user on a Windows Instance:

1. Log in to your instance using a Remote Desktop client.
2. On the **Start** menu, click **Control Panel**.
3. Click **User Accounts**, and then click **User Accounts** again.
4. Click **Manage User Accounts**.
5. Click **Manage Another Account**.
6. Click **Add User Account**.
7. Enter a **User name** and **Password**.
8. Confirm the password, and then create a **Password hint**.
9. Click **Next**.
10. Verify the account, and then click **Finish**.

You can now log in as the new user.

Displaying the Console for an Instance

You can capture and display the serial console data for an instance. The data includes configuration messages that occur when the instance boots, such as kernel and BIOS messages, and is useful for checking the status of the instance or diagnosing problems. Note that the *raw* console data, including multi-byte characters, is captured.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let users launch Compute instances includes the ability to manage console history data. If the specified group doesn't need to launch instances or attach volumes, you could simplify that policy to include only `manage instance-family`, and remove the statements involving `volume-family` and `virtual-network-family`.

Using the API

For information about using the API and signing requests, see REST APIs.

Use these API operations to manage the serial console logs:

- CaptureConsoleHistory
- DeleteConsoleHistory
- GetConsoleHistory
- GetConsoleHistoryContent
- ListConsoleHistories

Getting Instance Metadata

The metadata for an instance includes its OCID, display name, compartment, shape, region, availability domain, creation date, state, image, and any custom metadata that you provide, such as an SSH public key.

You can find some of this information in the Console on the **Compute** page, or you can get all of it by logging in to the instance and using the metadata service. The service runs on every instance and is an HTTP endpoint listening on 169.254.169.254.

Required IAM Policy

No IAM policy is required if you're logged in to the instance and using Curl to get the metadata (see below).

For administrators: Users can also get instance metadata through the Compute API (e.g., with GetInstance). The policy in Let users launch Compute instances covers that ability. If the

specified group doesn't need to launch instances or attach volumes, you could simplify that policy to include only `manage instance-family`, and remove the statements involving `volume-family` and `virtual-network-family`.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Accessing Instance Metadata on Oracle-Provided Images

You can get instance metadata on Oracle-provided images by using curl on Linux instances or using an Internet browser for Windows instances.

Using Curl to Get Linux Instance Metadata

After you connect to an instance using SSH, issue any of the following GET requests. You'll get back a response that includes all of the instance information, only the custom metadata, or only the custom metadata for the specified key name, respectively.

```
curl -L http://169.254.169.254/opc/v1/instance/
```

```
curl -L http://169.254.169.254/opc/v1/instance/metadata/
```

```
curl -L http://169.254.169.254/opc/v1/instance/metadata/<key-name>
```

In the example *<key-name>*, is `ssh_authorized_keys`, `user_data`, or any custom key name that you provided when you launched the instance. (For information about using the Core Services API to provide `user_data` to Cloud-Init, see LaunchInstanceDetails.)

The following example shows all of the information for an instance.

```
{
  "id" : "ocid1.instance.oc1.phx.abyhqljrkfpg67546xizk4welg3n4yft4hkud6hrdj5tietdnt7s4inffjoq",
  "displayName" : "rprods",
  "compartmentId" :
"ocid1.compartment.oc1..aaaaaaaal3gzijdhqol2pglie6astxxeyqdqeyg35nz5zxil2rggnx7jnhwa",
  "shape" : "x5-2.36.512",
  "region" : "phx",
  "availabilityDomain" : "cumS:PHX-AD-1",
```

```
  "timeCreated" : "2016-04-18T20:02:38.383+0000",
  "state" : "RUNNING",
  "image" : null,
  "metadata": {
      "ssh_authorized_keys" : "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAACAQCZ06fccNTQfq+JNL
       xubFlJ5ZR3kt+uzspdH9tXL+lAejSM1NXPMZKQEo73rKpUUkN121BqL46yTI2JjfFwjJWMJlTkFo
       M+CFZev7MIxfEjas06y80ZBZ7DUTQO0GxJPeD8NCOb1VorF8M4xuLwrmzRtkoZzU16umt4y1W0Q4
       ifdp3IiiU0U8/WxczSXcUVZOLqkz5dc6oMHdMVpkimietWzGZ4LBBsH/LjEVY7E0V+a0sNchlVDI
       Zcm7ErReBLcdTGDq0uLBiuChyl6RUkX1PNhusquTGwK7zc8OBXkRuubn5UKXhI3Ul9Nyk4XESkVW
       IGNKmw8mSpoJSjR8P9ZjRmcZVo8S+x4KVPMZKQEo73rKpUUkN121BqL46yTI2JjfFwjJWMJlTkFo
       EjRVJ/jf4IythUnkW5RA/2mgu79kbwqPM90J8pRKyjWehl8VsN5wUY+mZQ3jzIfeC9qNKjn5e976
       4DFhvw35JHh5sHyzyLVuyLe3teZ6kRUKyQqA9Oy4DMctmbD1uDAz73tWbvdz7SmfWJ5QZ7/Aod0a
       Gce6FJS/srWfB+7f/+SX+fu926LqlJa/3r/AGS4IvDfEKvtZCWgTPVrEHVSTuEiDzG9yBuJLZ95J
       2AMmeKhnFOKpAsoWVN5kV55RmmQVJaozQHr7V+FaGc8IHDs95vgz4F3AJTi+xl3cvr+6vlfDJNse
       c/jRx/+XZynrpltFGvTAUaqXJFowow== it.helps@company.com",
    "user_data" : "SWYgeW91IGNhbiBzZWUgdGhpcywgdGhlbiBZdCB3b3JrZWQgbWF5YmUuCg=="
}
```

Using an Internet Browser to Get Windows Instance Metadata

After you connect to a Windows instance, you can open an Internet browser such as Microsoft Edge or Internet Explorer, Google Chrome, or Mozilla Firefox, and then navigate to the following URLs:

- http://169.254.169.254/opc/v1/instance/
- http://169.254.169.254/opc/v1/instance/metadata/
- http://169.254.169.254/opc/v1/instance/metadata/*<key-name>*

In the example *<key-name>*, is `user_data` or any custom key name that you provided when you launched the instance.

Updating Instance Metadata

The Oracle Cloud Infrastructure Compute service lets you add and update custom metadata for an instance using the Command Line Interface (CLI) or REST APIs.

When you create an instance using the LaunchInstance operation you can specify custom metadata for the instance in the LaunchInstanceDetails datatype's `metadata` or `extendedMetadata` attributes. To update an instance's metadata, use the UpdateInstance operation, specifying the custom metadata in the UpdateInstanceDetails datatype's `metadata` or `extendedMetadata` attributes. The `metadata` attribute supports key/value string pairs while the `extendedMetadata` attribute supports nested JSON objects.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let users launch Compute instances includes the ability to rename an instance. If the specified group doesn't need to launch instances or attach volumes, you could simplify that policy to include only `manage instance-family`, and remove the statements involving `volume-family` and `virtual-network-family`.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Using the API

When you use the `UpdateInstance` operation, the instance's metadata will be the combination of the values specified in the UpdateInstanceDetails datatype's `metadata` or `extendedMetadata` attributes. Any set of key/value pairs specified for these attributes in the `UpdateInstance` operation will replace the existing values for these attributes, so you need to include all the metadata values for the instance in each call, not just the ones you want to add. If you leave the attribute empty when calling `UpdateInstance`, the existing metadata values in that attribute will be used. You cannot specify a value for the same metadata key twice, this will cause the `UpdateInstance` operation to fail due to there being duplicate keys.

To understand this, consider the example scenario where you created an instance using the `LaunchInstance` operation and specified the following key/value pair for the `metadata` attribute:

```
"myCustomMetadataKey" : "myCustomMetadataValue"
```

If you then call the `UpdateInstance` operation, and add new metadata by specifying additional key/value pairs in the `extendedMetadata` attribute, but you leave the `metadata` attribute empty, do not include the `myCustomMetadataKey` key/value in the `extendedMetadata` attribute, as this will cause the operation to fail since that key already exists. If you do specify values for the metadata attribute, you need to include the `myCustomMetadataKey` key/value to maintain it in the instance's metadata. In this case, you can specify it in either of the attributes.

There are two reserved keys, `user_data` and `ssh_authorized_keys`, that can only be set for an instance at launch time, they cannot be updated later. If you use the metadata attribute to add or update metadata to an instance, you need to ensure that you include the values specified at launch time for both these keys, otherwise the `UpdateInstance` operation will fail.

Best Practices for Updating an Instance's Metadata

When using the `UpdateInstance` operation, Oracle recommends the following:

- Use the GetInstance operation to retrieve the existing custom metadata for the instance to ensure that you include the values you want to maintain in the appropriate attributes when you call `UpdateInstance`. The metadata values are returned in the `metadata` and `extendedMetadata` attributes for the Instance . For a code example demonstrating this, see the UpdateInstanceExample in the SDK for Java.
- Unless you are updating custom metadata that was added using the `metadata` attribute, use the `extendedMetadata` attribute to add custom metadata. Otherwise you need to include the launch time values for the `user_data` and `ssh_authorized_keys` reserved keys. If you use the `metadata` attribute to add values and you leave out the values for these reserved keys or specify different values for them, the `UpdateInstance` call will fail.

Renaming an Instance

You can rename an instance without changing its Oracle Cloud Identifier (OCID).

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let users launch Compute instances includes the ability to rename an instance. If the specified group doesn't need to launch instances or attach volumes, you could simplify that policy to include only `manage instance-family`, and remove the statements involving `volume-family` and `virtual-network-family`.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the UpdateInstance operation to change the name of an instance.

Moving a Compute Instance

This topic covers how to relocate a virtual machine or a bare metal instance by using reboot migration or a manual process.

Reboot Migration

For instances with a date in the **Reboot Maintenance** field (available in the Console, CLI, and SDKs), you can reboot your instance to move it to new infrastructure. After you reboot the instance, the **Reboot Maintenance** field is cleared. This change indicates that the instance was moved successfully.

Before the reboot, detach all paravirtualized or emulated block volumes and delete all secondary VNIC attachments. Left in place, these attachments can prevent the service from moving your instance. After the reboot, you can reattach block volumes and secondary VNICs.

Limitations and Warnings for Reboot Migration

Any public IP addresses assigned to secondary VNICs from a reserved public pool are retained. Public IP addresses that were not assigned from a reserved public IP pool will change. Private IP addresses do not change.

Prerequisites for Reboot Migration

1. Determine if your instance has any paravirtualized or emulated remote block volume attachments:
    a. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
    b. Click the name of the instance you want to move.
    c. On the **Instance Details** page, click **Attached Block Volumes**.
    d. Note the **Attachment Type** for each volume.
2. Document any remote block volumes attached to the instance.
3. Determine if your instance has any secondary VNIC attachments:
    a. On the **Instance Details** page, click **Attached VNICs**.

b. Excluding the first VNIC, document all private IP addresses, names, subnets, and tags on each VNIC. Note that the instance can have multiple VNICs, and each VNIC can have multiple secondary IP addresses.

c. Excluding the first VNIC, document any public IP addresses assigned from a reserved public pool. Note that the instance can have multiple VNICs, and each VNIC can have multiple secondary private IP addresses. Each VNIC and secondary private IP address can have an attached public IP address.

4. Prepare the instance for reboot migration:
   - Ensure that any remote block volumes defined in `/etc/fstab` use the recommended options.
   - Ensure that any File Storage service (NFS) mounts use the `nofail` option.
   - If you have statically defined any network interfaces belonging to secondary VNICs using their MAC addresses, such as those defined in `/etc/sysconfig/network-scripts/ifcfg*`, those interfaces will not start due to the change in the MAC address. Remove the static mapping.
   - If you use the Oracle-provided script to configure secondary VNICs, ensure it runs automatically at startup.

Moving an Instance with Reboot Migration

After you complete the prerequisites:

1. Stop any running applications.
2. Ensure that those applications will not start automatically.



Warning

When the relocated instance starts for the first time, remote block volumes and secondary VNICs, or any resource that relies on them, will not be attached. The absence of these resources can cause application issues.

3. Unmount any remote block volumes or File Storage service (NFS) mounts.
4. Detach any paravirtualized or emulated remote block volumes.
5. Delete all secondary VNICs.
6. Back up all remote block volumes. See Backing Up a Volume for more information.
7. Reboot the instance.
8. Confirm that the **Reboot Maintenance** field no longer has a date.
9. Reattach any remote block volumes.
10. Recreate any secondary VNICs, including tags.
11. Start and test any applications on the instance.
12. Configure the applications to start automatically, as required.
13. (Optional) After you confirm that the instance and applications are healthy, you can delete the volume backups.

Moving an Instance with Manual Migration

For instances without a date in the **Reboot Maintenance** field (available in the Console, CLI, and SDKs), you must move the instance manually. This method requires that you terminate the instance, and then launch a new instance from the retained boot volume. Instances that have additional VNICs, secondary IP addresses, remote attached block volumes, or that belong to a backend set of a load balancer require additional steps.

Limitations and Warnings for Manual Migration

Be aware of the following limitations and warnings when performing a manual migration:

- Any public IP addresses assigned to your instance from a reserved public pool are retained. Any that were not assigned from a reserved public IP address pool will change. Private IP addresses do not change.
- MAC addresses, CPUIDs, and other unique hardware identifiers do change during the move. If any applications running on the instance use these identifiers for licensing or other purposes, be sure to take note of this information before moving the instance to help you manage the change.

Prerequisites for Manual Migration

1. Before moving the instance, document all critical details:
    - The instance's region, availability domain, and fault domain.
    - The instance's display name.
    - All private IP addresses, names, and subnets. Note that the instance can have multiple VNICs, and each VNIC can have multiple secondary IP addresses.
    - All private DNS names. The instance can have multiple VNICs, and each VNIC can have multiple secondary IP addresses. Each private IP address can have a DNS name.
    - Any public IP addresses assigned from a reserved public pool. Note that the instance can have multiple VNICs, and each VNIC can have multiple secondary private IP addresses. Each VNIC and secondary private IP address can have an attached public IP address.
    - Any remote block volumes attached to the instance.
    - Any tags on the instance or attached resources.
2. Prepare the instance for manual migration:
    - Ensure that any remote block volumes defined in `/etc/fstab` use the recommended options.

- Ensure that any File Storage service (NFS) mounts use the `nofail` option.
- If you have statically defined any network interfaces belonging to secondary VNICs using their MAC addresses, such as those defined in `/etc/sysconfig/network-scripts/ifcfg*`, those interfaces will not start due to the change in the MAC address. Remove the static mapping.
- If you use the Oracle-provided script to configure secondary VNICs, ensure it runs automatically at startup.

Moving an Instance Manually

After you complete the prerequisites:

1. Stop any running applications.
2. Ensure that those applications will not start automatically.



Warning

When the relocated instance starts for the first time, remote block volumes, secondary VNICs, or any resource that relies on them, will not be attached. The absence of these resources can cause application issues.

3. If your instance has local NVMe storage (dense instances), you must back up this data:
   a. Create and attach one or more remote block volumes to the instance.
   b. Copy the data from the NVMe devices to the remote block volumes.
4. Unmount any remote block volumes or File Storage service (NFS) mounts.
5. Back up all remote block volumes. See Overview of Block Volume Backups for more information.
6. Create a backup of the root volume.



Important

Do not generalize or specialize Windows instances.

7. Terminate the instance:

Using the Console

a. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.

b. In the list of instances, find the instance you want to terminate.

c. Click the highlighted name of the instance to display the instance details.

d. Click **Terminate**, and then respond to the confirmation prompt. Ensure that `Permanently delete the attached Boot Volume` is unchecked to preserve the boot volume associated with the instance.
Terminated instances temporarily remain in the list of instances with the status **Terminated**.

Using the API

To terminate the instance, use the TerminateInstance operation and pass the `preserveBootVolume` parameter set to `true` in the request.

Using the CLI

To terminate the instance, use the terminate operation and set the `preserve-boot-volume` option to `true`.

8. Create a new instance using the boot volume from the terminated instance.

9. In the launch instance flow, specify the private IP address that was attached to the primary VNIC. If the public IP address was assigned from a reserved IP address pool, be sure to assign the same IP address.
10. When the instance state changes to `RUNNING`, **Stop** the instance.
11. Recreate any secondary VNICs and secondary IP addresses.
12. Attach any remote block volumes.



Note

This step includes any volumes used to back up local NVMe devices. Copy the data onto the NVMe storage on the new instance, and then detach the volumes.

13. Start the instance.
14. Start and test any applications on the instance.
15. Configure the applications to start automatically, as required.
16. Recreate the required tags.
17. (Optional) After you confirm that the instance and applications are healthy, you can delete the volume backups.

Migrating an Instance from a Local to Remote Boot Volume

Oracle Cloud Infrastructure introduced remote boot volumes November 15th, 2017. Remote boot volumes include significant improvements over local boot volumes including lower boot times, encryption of data at rest and in-transit, zero downtime snapshots, improved availability, and more. Instances launched prior to November 15th, 2017 use local boot volumes. Local boot volumes are being deprecated, we recommend that you migrate any instances currently using a local boot volume to a remote boot volume. This topic describes that process.

Limitations and Warnings for Manual Migration

Be aware of the following limitations and warnings when performing a manual migration:

- Any public IP addresses assigned to your instance from a reserved public pool are retained. Any that were not assigned from a reserved public IP address pool will change. Private IP addresses do not change.
- MAC addresses, CPUIDs, and other unique hardware identifiers do change during the move. If any applications running on the instance use these identifiers for licensing or other purposes, be sure to take note of this information before moving the instance to help you manage the change.

Prerequisites for Manual Migration

1. Before moving the instance, document all critical details:
    - The instance's region, availability domain, and fault domain.
    - The instance's display name.
    - All private IP addresses, names, and subnets. Note that the instance can have multiple VNICs, and each VNIC can have multiple secondary IP addresses.
    - All private DNS names. The instance can have multiple VNICs, and each VNIC can have multiple secondary IP addresses. Each private IP address can have a DNS name.
    - Any public IP addresses assigned from a reserved public pool. Note that the instance can have multiple VNICs, and each VNIC can have multiple secondary private IP addresses. Each VNIC and secondary private IP address can have an attached public IP address.
    - Any remote block volumes attached to the instance.
    - Any tags on the instance or attached resources.
2. Prepare the instance for manual migration:
    - Ensure that any remote block volumes defined in `/etc/fstab` use the recommended options.

- Ensure that any File Storage service (NFS) mounts use the `nofail` option.
- If you have statically defined any network interfaces belonging to secondary VNICs using their MAC addresses, such as those defined in `/etc/sysconfig/network-scripts/ifcfg*`, those interfaces will not start due to the change in the MAC address. Remove the static mapping.
- If you use the Oracle-provided script to configure secondary VNICs, ensure it runs automatically at startup.

Migrating an Instance Manually

After you complete the prerequisites:

1. Stop any running applications.
2. Ensure that those applications will not start automatically.



Warning

When the relocated instance starts for the first time, remote block volumes, secondary VNICs, or any resource that relies on them, will not be attached. The absence of these resources can cause application issues.

3. If your instance has local NVMe storage (dense instances), you must back up this data:
   a. Create and attach one or more remote block volumes to the instance.
   b. Copy the data from the NVMe devices to the remote block volumes.
4. Unmount any remote block volumes or File Storage service (NFS) mounts.
5. Back up all remote block volumes. See Overview of Block Volume Backups for more information.
6. Create a custom image of the instance using the steps described in the Using the Console or Using the API sections of Managing Custom Images.



Important

Do not generalize or specialize Windows instances.

7. Terminate the instance:

Using the Console

a. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.

b. In the list of instances, find the instance you want to terminate.

c. Click the highlighted name of the instance to display the instance details.

d. Click **Terminate**, and then respond to the confirmation prompt. Ensure that `Permanently delete the attached Boot Volume` is unchecked to preserve the boot volume associated with the instance.
Terminated instances temporarily remain in the list of instances with the status **Terminated**.

Using the API

To terminate the instance, use the TerminateInstance operation and pass the `preserveBootVolume` parameter set to `true` in the request.

Using the CLI

To terminate the instance, use the terminate operation and set the `preserve-boot-volume` option to `true`.

8. Create a new instance using the custom image from the terminated instance, see the steps described in the Using the Console or the Using the API sections of Managing Custom Images.

9. In the launch instance flow, specify the private IP address that was attached to the primary VNIC. If the public IP address was assigned from a reserved IP address pool, be sure to assign the same IP address.
10. When the instance state changes to `RUNNING`, **Stop** the instance.
11. Recreate any secondary VNICs and secondary IP addresses.
12. Attach any remote block volumes, see Attaching a Volume for more information.



Note

This step includes any volumes used to back up local NVMe devices. Copy the data onto the NVMe storage on the new instance, and then detach the volumes.

13. Start the instance.
14. Start and test any applications on the instance.
15. Configure the applications to start automatically, as required.
16. Recreate the required tags.
17. (Optional) After you confirm that the instance and applications are healthy, you can delete the volume backups.

Stopping and Starting an Instance

You can stop and start an instance as needed to update software or resolve error conditions.

Stopping or Restarting an Instance From Within the Instance

In addition to using the API and Console, you can stop and restart instances using the commands available in the operating system when you are logged in to the instance. Stopping an instance from within the instance does not stop billing for that instance. If you stop an instance this way, be sure to also stop it from the Console or API.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let users launch Compute instances includes the ability to stop or start an existing instance. If the specified group doesn't need to launch instances or attach volumes, you could simplify that policy to include only `manage instance-family`, and remove the statements involving `volume-family` and `virtual-network-family`.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Rebooting Your Virtual Machine (VM) Instance During Planned Maintenance

When an underlying Oracle Cloud Infrastructure component needs to undergo maintenance, you are notified in advance of the impact to your VM instances. If your VM instances are scheduled for a maintenance reboot, you can pro-actively reboot, or stop and start your instances using the Console, API, or CLI at any time before the scheduled reboot. This allows you to control how and when your applications experience downtime.

To identify VM instances that can be pro-actively rebooted using the Console, check the **Maintenance Reboot** field for the instance. If the instance has a maintenance reboot scheduled and can be proactively rebooted, this field displays the date and start time for the reboot. To check this using the API, use the `timeMaintenanceRebootDue` field for the Instance. For VM instances with a boot volume, additional iSCSI block volume attachments, and a single VNIC, you can proceed to reboot, or stop and start the instance. If you have non-iSCSI (paravirtualized or emulated) block volume attachments or secondary VNICs, you need to first detach these resources before you reboot your instance.

When you reboot, or stop and start the instance, it will be migrated to a different physical VM host. Once the **Maintenance Reboot** field is blank, the instance will no longer be impacted by the maintenance event. If you choose not to reboot before the scheduled time, then Oracle Cloud Infrastructure will reboot and migrate your instances within a 24-hour period after the scheduled time.

To make it easier to locate and perform these actions on your VM instances, you can use Search with a predefined query to find all instances that have a maintenance reboot scheduled.

To find all the instances scheduled for a maintenance reboot

1. In the Console, append "/a/query" to the end of your base Console URL. For example, https://console.us-ashburn-1.oraclecloud.com/a/query.
2. Click **Select Sample Query**, and then click **Query for all instances which have an upcoming scheduled maintenance**.



Note

The **Maintenance Reboot** field is supported on all standard VM instances running Linux-based operating systems. For VM instances running other operating systems, follow the instructions in the Oracle Cloud Infrastructure maintenance notification email.

Resource Billing for Stopped Instances

For both VM and bare metal instances, billing depends on the shape that you use to create the instance:

- **Standard shapes:** Stopping an instance pauses billing. However, stopped instances continue to count toward your service limits.

- **Dense I/O shapes:** Billing continues for stopped instances because of the attached NVMe storage, and related resources continue to count toward your service limits. To halt billing and remove related resources from your service limits, you must terminate the instance.
- **GPU shapes:** Stopping an instance pauses billing. However, stopped instances continue to count toward your service limits.
- **HPC shapes:** Billing continues for stopped instances because of the attached NVMe storage, and related resources continue to count toward your service limits. To halt billing and remove related resources from your service limits, you must terminate the instance.

Stopping an instance from within the instance does not stop billing for that instance. If you stop an instance this way, be sure to also stop it from the Console or API.

Using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
2. In the list of instances, find the instance you want to stop or start, and then click the instance name to display the instance details.
3. Click one of the following actions:

START

Restarts a stopped instance. After the instance is restarted, the **Stop** action is enabled.

STOP

Shuts down the instance. After the instance is powered off, the **Start** action is enabled.

REBOOT

Shuts down the instance, and then restarts it.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the InstanceAction operation to restart an instance.

The following actions are only available using the API:

- SOFTSTOP
- SOFTRESET

Terminating an Instance

You can permanently terminate (delete) instances that you no longer need. Any attached VNICs and volumes are automatically detached when the instance terminates. Eventually, the instance's public and private IP addresses are released and become available for other instances. By default, the instance's boot volume is deleted when you terminate the instance, however you can preserve the boot volume associated with the instance, so that you can attach it to a different instance as a data volume, or use it to launch a new instance.



Warning

If your instance has NVMe storage, terminating it securely erases the NVMe drives and the data that was on those drives becomes completely unrecoverable. Make sure you back up important data before terminating an instance. For more information, see Protecting Data on NVMe Devices.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let users launch Compute instances includes the ability to terminate an instance (with or without an attached block volume).

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for instances, cloud networks, or other Core Services API resources, see Details for the Core Services.

Using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
2. In the list of instances, find the instance you want to terminate.
3. Click the highlighted name of the instance to display the instance details.
4. Click **Terminate**, and then respond to the confirmation prompt.
   If you want to preserve the boot volume associated with the instance, uncheck `Permanently delete the attached Boot Volume`.
   Terminated instances temporarily remain in the list of instances with the status **Terminated**.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the TerminateInstance operation to terminate an instance.

Enabling Monitoring for Compute Instances

This topic describes how to enable monitoring for Compute instances that use supported images.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Except for Compute instances, all resources that support the Monitoring service emit metrics by default. A Compute instance using a supported image emits metrics when it has the required instance configuration and OracleCloudAgent software.

Supported Images

Only Compute instances with supported images provide the configuration setting to enable monitoring. When this setting is enabled for an instance and the OracleCloudAgent software is installed on the instance, the instance emits metrics.



Note

Because legacy images require installation of the OracleCloudAgent software, we recommend selecting the latest image, which already has the software installed. If you need a legacy image, then select an image dated on or after November 16, 2018 (except Ubuntu, which must be a new image).

- CentOS 6 (named CentOS-6.*x*-<*date*>-<*number*>)
- CentOS 7 (named CentOS-7-<*date*>-<*number*>)
- Oracle Linux 6 (Oracle Linux 6 Unbreakable Enterprise Kernel Release 4, named Oracle-Linux-6.*x*-<*date*>-<*number*>)
- Oracle Linux 7.x (Oracle Linux 7 Unbreakable Enterprise Kernel Release 4, named Oracle-Linux-7.*x*-<*date*>-<*number*)
- Ubuntu 14.04 (named Canonical-Ubuntu-14.04-<*date*>-<*number*>)
- Ubuntu 16.04 (named Canonical-Ubuntu-16.04-<*date*>-<*number*>)
- Ubuntu 18.04 (named Canonical-Ubuntu-18.04-<*date*>-<*number*>)
- Windows Server 2008 R2 (named Windows-Server-2008-R2-Enterprise-Edition-VM-<*date*>-<*number*)
- Windows Server 2012 R2 (named Windows-Server-2012-R2-<*edition*>-<*gen*>.<*date*>-<*number*)
- Windows Server 2016 (named Windows-Server-2016-<*edition*>-Gen2.<*date*>-<*number*>)

Prerequisites

- Service gateways or public IP addresses: The Compute instances must have either service gateways or public IP addresses to send metrics to the Monitoring service.
- IAM policies: To create and update Compute instances, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in. For more information, see Creating an Instance.
- SSH: To connect to a Compute instance, you must have an SSH key. For more information, see Connecting to an Instance.

Process Overview: Enabling Monitoring for a New Compute Instance

Following is the process for configuring a new Compute instance to emit metrics.

Task 1: Create a monitoring-enabled instance

Steps depend on the date of the image used to create the instance.

Latest version of supported image



Note

Like the latest version, some recent versions of supported images also have the OracleCloudAgent software installed. Compare to the date listed in Supported Images.

While defining properties for the new instance, set the property that enables the instance for monitoring. For instructions, see Using the Console or Using the API.

Legacy version of supported image

A legacy version of a supported image is one provided before the date listed in Supported Images.

While defining properties for the new instance, set the property that enables the instance for monitoring and do one of the following:

- In the Console: While defining properties for the new instance, provide a script to install the OracleCloudAgent software onto the instance during the instance creation process.
- Complete the instance creation process and then manually install the OracleCloudAgent software.

Task 2: (Optional) Create a service gateway

If your instances do not have public IP addresses, set up a service gateway on the virtual cloud network (VCN). The service gateway allows the instances to send metrics to the Monitoring service without the traffic going over the internet. Here are special notes for setting up the service gateway to access the Monitoring service:

- When creating the service gateway, enable the service label called **All <region> Services in Oracle Services Network**. It includes the Monitoring service.
- When setting up routing for the subnet that contains your instances, set up a route rule with **Target Type** set to **Service Gateway**, and the **Destination Service** set to **All <region> Services in Oracle Services Network**.

For detailed instructions, see Setting Up a Service Gateway in the Console.

Find Out if Monitoring Has Your Metrics

To find out if Monitoring has your metrics

This task involves either querying instance metrics or viewing instance configuration.

- Query metrics to determine if Monitoring is receiving metrics emitted by the instance. For instructions, see Using the Console or Using the API.

Not seeing metrics for your instance?

If you don't see any metric charts, your Compute instance might not be emitting metrics. See the following possible causes and resolutions.

Possible cause	How to check	Resolution
Monitoring is disabled on the instance.	Review the instance configuration.	Enable monitoring.
No OracleCloudAgent software exists on the instance (occurs with older images).	Connect to the instance and look for the software.	Install the software.
The instance cannot access theMonitoring service because its VCN does not use the Internet.	Review the instance's IP address. If it's not public, then a service gateway is needed.	Set up a service gateway.

Possible cause	How to check	Resolution
The instance does not use a supported image.	Review Supported Images.	Create an instance with a supported image.
New instance in a new compartment: The IAM policies required for the instance to publish metrics to Monitoring are not yet initialized. More information: IAM policies are automatically created for new instances and are immediately available, unless the instances are in a new compartment. For a new instance in a new compartment, the policies can take up to 20 minutes to initialize, which delays the emission of metrics.	(not applicable)	Check back after 10 or 20 minutes.

- View the instance configuration to determine if monitoring is enabled. Monitoring-enabled instances may require installation of the OracleCloudAgent software before metrics are emitted. For instructions, see Using the Console or Using the API.

Process Overview: Enabling Monitoring for an Existing Compute Instance

Following is the process for configuring an existing Compute instance to emit metrics.

Task 1: Enable monitoring

Update the instance configuration to enable monitoring. For instructions, see Using the Console or Using the API.

Task 2: Install the OracleCloudAgent software

Choose the operating system corresponding to the instance.

Linux

This section covers CentOS, Linux, and Ubuntu images.

1. Enable monitoring on the instance.
2. Connect to the instance.
   For step-by-step instructions, see Connecting to an Instance.
3. Run the script corresponding to the image used by the instance.

CentOS 6.x, Oracle Linux 6.x

```
#!/bin/sh

cd ~
curl -O https://objectstorage.us-phoenix-
1.oraclecloud.com/p/bUGHSOrPjZalGUX3Lmpe7BS6oKzEQF86XuIFwOJVXAs/n/imagegen/b/agents/o/oracle-
cloud-agent-0.0.8-176.el6.x86_64.rpm -v
```

CentOS 7.x, Oracle Linux 7.x

```
#!/bin/sh

cd ~
curl -O https://objectstorage.us-phoenix-1.oraclecloud.com/p/IX8WN_S_
AJtA4kcThoambuQ4nYkim4Z7Ml1mW9l_nIo/n/imagegen/b/agents/o/oracle-cloud-agent-0.0.8-176.el7.x86_
64.rpm -v
```

Ubuntu 14.04, 16.04, 18.04



Note

Installation of the OracleCloudAgent software on instances using Ubuntu images requires Snapcraft. To install Snapcraft, run the following commands, in sequence:

```
sudo apt update
```

```
sudo apt install snapd
```

```
sudo snap install oracle-cloud-agent --classic
```

4. Enter the relevant command to run the OracleCloudAgent software on the instance.

Example 1: CentOS or Oracle Linux image

```
sudo yum install -y <instance-agent-filename>
```

Example 2: Ubuntu image

(Same instructions as noted in the previous step: This command installs and runs the software.)

```
sudo snap install oracle-cloud-agent --classic
```

Metrics are now emitted by the instance.

Windows

1. Enable monitoring on the instance.

2. Download the OracleCloudAgent software from the following URL.
   https://objectstorage.us-phoenix-1.oraclecloud.com/p/RDDCRrNTo5WB19mI52QJTOXCq1whXFZHXCQP74b4ttg/n/imagegen/b/windows_instance_agents/o/OracleCloudAgentSetup.msi
3. Connect to the instance.
   For step-by-step instructions, see Connecting to an Instance.
4. Copy the downloaded OracleCloudAgent software to the instance.
5. Enter the relevant command to run the OracleCloudAgent software on the instance.

   ```
   msiexec /i /s <instance-agent-filename>
   ```

   Metrics are now emitted by the instance.

Task 3: (Optional) Create a service gateway

If your instances do not have public IP addresses, set up a service gateway on the virtual cloud network (VCN). The service gateway allows the instances to send metrics to the Monitoring service without the traffic going over the internet. Here are special notes for setting up the service gateway to access the Monitoring service:

- When creating the service gateway, enable the service label called **All <region> Services in Oracle Services Network**. It includes the Monitoring service.
- When setting up routing for the subnet that contains your instances, set up a route rule with **Target Type** set to **Service Gateway**, and the **Destination Service** set to **All <region> Services in Oracle Services Network**.

For detailed instructions, see Setting Up a Service Gateway in the Console.

Using the Console

Use the Console to create a Compute instance with monitoring enabled and, for new instances using legacy images, to run the script that installs the required OracleCloudAgent software.

To create a monitoring-enabled instance

Steps depend on the date of the image used to create the instance.

Latest version of supported image



Note

Like the latest version, some recent versions of supported images also have the OracleCloudAgent software installed. Compare to the date listed in Supported Images.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**. Choose a **Compartment** you have permission to work in, and then click **Create Instance**.
2. In the **Create Instance** dialog box, select the latest version of a supported image. For more information about launching instances, see Creating an Instance.
3. Select **Enable monitoring**.
4. Update other configuration as needed and then click **Create Instance**. The newly created monitoring-enabled instance emits metrics to the Monitoring service.

Legacy version of supported image

A legacy version of a supported image is one provided before the date listed in Supported Images. Legacy images require you to install the OracleCloudAgent software.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**. Choose a **Compartment** you have permission to work in, and then click

Create Instance.

2. In the **Create Instance** dialog box, select the legacy version of the supported image. For more information about launching instances, see Creating an Instance.
3. Select **Enable monitoring**.
4. Click **Show Advanced Options**.
5. Select **Paste cloud-init script**.
6. Copy and paste the script corresponding to the image used by the instance.

CentOS 6.x, Oracle Linux 6.x

```
#!/bin/sh
curl -O https://objectstorage.us-phoenix-
1.oraclecloud.com/p/bUGHSOrPjZalGUX3Lmpe7BS6oKzEQF86XuIFwOJVXAs/n/imagegen/b/agents/o/oracle-
cloud-agent-0.0.8-176.el6.x86_64.rpm
yum install -y ~/oracle-cloud-agent-0.0.8-176.el6.x86_64.rpm -v
```

CentOS 7.x, Oracle Linux 7.x

```
#!/bin/sh
curl -O https://objectstorage.us-phoenix-1.oraclecloud.com/p/IX8WN_S_
AJtA4kcThoambuQ4nYkim4Z7Ml1mW9l_nIo/n/imagegen/b/agents/o/oracle-cloud-agent-0.0.8-176.el7.x86_
64.rpm -v
yum install -y ~/oracle-cloud-agent-0.0.8-176.el7.x86_64.rpm -v
```

Windows Server 2012 R2, 2016



Note

For legacy versions of Windows images, make sure cloudbase-init is supported. See https://docs.cloud.oracle.com/iaas/releasenotes/changes/595afbb7-de0c-4934-8074-5b1ed6be1b56/.

```
#ps1_sysnative
cd \Users\opc\Desktop
Start-BitsTransfer -Source "https://objectstorage.us-phoenix-
1.oraclecloud.com/p/RDDCRrNTo5WB19mI52QJTOXCq1whXFZHXCQP74b4ttg/n/imagegen/b/windows_instance_
agents/o/OracleCloudAgentSetup.msi" -Destination
"c:\Users\opc\Desktop\OracleCloudAgentSetup.msi"
msiexec /i "c:\Users\opc\Desktop\OracleCloudAgentSetup.msi" /quiet /L*V
"c:\Users\opc\Desktop\OracleCloudAgentSetup.log"
```

Windows Server 2008 R2

Download the OracleCloudAgent software from the following URL and manually install it on the instance.

https://objectstorage.us-phoenix-1.oraclecloud.com/p/RDDCRrNTo5WB19mI52QJTOXCq1whXFZHXCQP74b4ttg/n/imagegen/b/windows_instance_agents/o/OracleCloudAgentSetup.msi

Ubuntu 14.04, 16.04, 18.04



Note

Installation of the OracleCloudAgent software on instances using Ubuntu images requires Snapcraft. To install Snapcraft, run the following commands, in sequence:

```
sudo apt update
```

```
sudo apt install snapd
```

```
sudo snap install oracle-cloud-agent --classic
```

7. Update other configuration as needed and then click **Create Instance**.
   The newly created monitoring-enabled instance emits metrics to the Monitoring service.

To find out if monitoring is enabled or if Monitoring is receiving metrics

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
2. Click the instance you're interested in.
3. On the instance detail page, under **Resources**, click **Metrics**.
   If you see metric charts with data, then the Monitoring service is receiving metrics from this instance. For a list of metrics related to Compute instances, see Compute Metrics.
   If you see a message that monitoring is not enabled, or that the OracleCloudAgent software needs to be installed, then complete those tasks.

To enable monitoring on an existing instance

1. Go to the Metrics page for the instance:
   a. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
   b. Click the instance you're interested in.
   c. On the instance detail page, under **Resources**, click **Metrics**.
      If monitoring is not enabled (and the instance uses a supported image), then a button is available for enabling monitoring.
2. Click **Enable monitoring**.
   If you see metric charts with data, then the Monitoring service is receiving metrics from this instance. For a list of metrics related to Compute instances, see Compute Metrics.
   If you see a message that the OracleCloudAgent software needs to be installed, then see Task 2: Install the OracleCloudAgent software.

Using the API

Use the API to enable monitoring on a new or existing instance. After monitoring is enabled, you can install the OracleCloudAgent software.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

To create a monitoring-enabled instance

Steps depend on the date of the image used to create the instance.

Latest version of supported image



Note

Like the latest version, some recent versions of supported images also have the OracleCloudAgent software installed. Compare to the date listed in Supported Images.

Use the LaunchInstance API operation, specifying the latest version of a supported image, and include the following parameter setting.

```
{
  "agentConfig":
  {
    "isMonitoringDisabled":false
  }
}
```

Legacy version of supported image

A legacy version of a supported image is one provided before the date listed in Supported Images. Legacy images require you to install the OracleCloudAgent software.

1. Use the LaunchInstance API operation, specifying the legacy version of the supported image, and include the following parameter setting.

```
{
  "agentConfig":
  {
    "isMonitoringDisabled":false
  }
}
```

2. Install the OracleCloudAgent software onto the newly created instance. See Task 2: Install the OracleCloudAgent software.

To find out if monitoring is enabled or if Monitoring is receiving metrics

To query metrics, use the SummarizeMetricsData API operation. Returned metrics indicate that the Monitoring service received metrics from the instance.

To determine instance agent configuration (`isMonitoringDisabled` value), use the GetInstance or ListInstances operation.

To enable monitoring on an existing instance

Use the UpdateInstance API operation and include the following parameter setting.

```
{
  "agentConfig":
  {
    "isMonitoringDisabled":false
  }
}
```

Compute Metrics

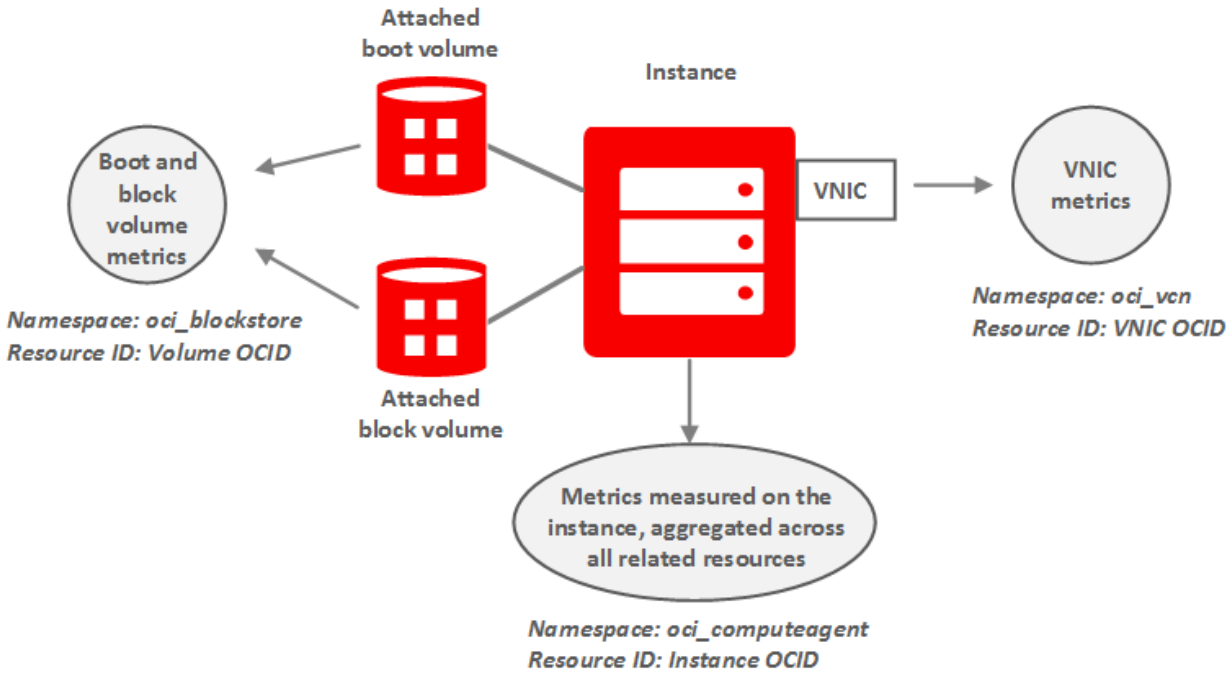
You can monitor the health, capacity, and performance of your Compute instances by using metrics, alarms, and notifications.

This topic describes the metrics emitted by the metric namespace `oci_computeagent` (the OracleCloudAgent software on the Compute instances).

Resources: Monitoring-enabled Compute instances.

Overview of Metrics for an Instance and Related Resources

This section gives an overall picture of the different types of metrics available for an instance and its storage and network devices. See the following diagram and table for a summary.



Metric Namespace	Resource ID	Where Measured	Available Metrics
`oci_computeagent`	Instance OCID	On the instance. The metrics in this namespace are aggregated across all the related resources on the instance. For example, `DiskBytesRead` is aggregated across all the instance's attached storage volumes, and `NetworkBytesIn` is aggregated across all the instance's attached VNICs.	See Available Metrics: oci_computeagent.
`oci_blockstore`	Boot or block volume OCID	By the Block Volume service. The metrics are for an individual volume (either boot volume or block volume).	See Block Volume Metrics
`oci_vcn`	VNIC OCID	By the Networking service. The metrics are for an individual VNIC.	See Networking Metrics

Prerequisites

- IAM policies: To monitor resources, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. The policy must give you access to the monitoring services as well as the resources being monitored. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in. For more information on user authorizations for monitoring, see the Authentication and Authorization section for the related service: Monitoring or Notifications.

- Metrics exist in Monitoring: The resources that you want to monitor must emit metrics to the Monitoring service.
- Compute instances: To emit metrics, Compute instances must be monitoring-enabled. OracleCloudAgent software installation may also be required. For more information, see Enabling Monitoring for Compute Instances.

Available Metrics: oci_computeagent

The Compute service metrics help you measure activity level and throughput of Compute instances. The metrics listed in the following table are available for any monitoring-enabled Compute instance. You must enable monitoring on the instances to get these metrics.

The metrics in this namespace are aggregated across all the related resources on the instance. For example, `DiskBytesRead` is aggregated across all the instance's attached storage volumes, and `NetworkBytesIn` is aggregated across all the instance's attached VNICs.

You also can use the Monitoring service to create custom queries.

Each metric includes the following dimensions:

AVAILABILITYDOMAIN

The availability domain where the instance resides.

FAULTDOMAIN

The fault domain where the instance resides.

IMAGEID

The OCID of the image for the instance.

INSTANCEPOOLID

The instance pool that the instance belongs to.

REGION

The region where the instance resides.

RESOURCEDISPLAYNAME

The friendly name of the instance.

RESOURCEID

The OCID of the instance.

SHAPE

The shape of the instance.

Metric	Metric Display Name	Unit	Description	Dimensions
CpuUtilization	**CPU Utilization**	percent	Activity level from CPU. Expressed as a percentage of total time.	availabilityDomain faultDomain imageId instancePoolId region resourceDisplayName resourceId shape
DiskBytesRead[1, 3]	**Disk Read Bytes**	bytes	Read throughput. Expressed as bytes read per interval.	
DiskBytesWritten[1, 3]	**Disk Write Bytes**	bytes	Write throughput. Expressed as bytes written per interval.	
DiskIopsRead[1, 3]	**Disk Read I/O**	operations	Activity level from I/O reads. Expressed as reads per interval.	

Metric	Metric Display Name	Unit	Description	Dimensions
DiskIopsWritten[1, 3]	**Disk Write I/O**	operations	Activity level from I/O writes. Expressed as writes per interval.	
MemoryUtilization[1]	**Memory Utilization**	percent	Space currently in use. Measured by pages. Expressed as a percentage of used pages.	

Metric	Metric Display Name	Unit	Description	Dimensions
NetworksBytesIn[1, 2]	**Network Receive Bytes**	bytes	Network receipt throughput. Expressed as bytes received.	
NetworksBytesOut[1, 2]	**Network Transmit Bytes**	bytes	Network transmission throughput. Expressed as bytes transmitted.	

[1]This metric is a cumulative counter that shows monotonically increasing behavior for each session of the OracleCloudAgent software, resetting when the operating system is restarted.

[2]The Networking service provides additional metrics (in the `oci_vcn` metric namespace) for each VNIC on the instance. For more information, see Networking Metrics.

[3]The Block Volume service provides additional metrics (in the `oci_blockstore` metric namespace) for each volume attached to the instance. For more information, see Block Volume Metrics.

Using the Console

To view default metric charts for a single Compute instance

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
2. In the list of instances, find the instance you want to view metric charts for, and then click the instance name to display the instance details.
3. Under **Resources**, click **Metrics**.
   The Metrics page displays a chart for each metric that is emitted by the metric namespace for Compute instances. For more information about the emitted metrics, see Compute Metrics.

Not seeing any metric charts for the instance?

If you don't see any metric charts, your Compute instance might not be emitting metrics. See the following possible causes and resolutions.

Possible cause	How to check	Resolution
Monitoring is disabled on the instance.	Review the instance configuration.	Enable monitoring.
No OracleCloudAgent software exists on the instance (occurs with older images).	Connect to the instance and look for the software.	Install the software.

Possible cause	How to check	Resolution
The instance cannot access theMonitoring service because its VCN does not use the Internet.	Review the instance's IP address. If it's not public, then a service gateway is needed.	Set up a service gateway.
The instance does not use a supported image.	Review Supported Images.	Create an instance with a supported image.
New instance in a new compartment: The IAM policies required for the instance to publish metrics to Monitoring are not yet initialized. More information: IAM policies are automatically created for new instances and are immediately available, unless the instances are in a new compartment. For a new instance in a new compartment, the policies can take up to 20 minutes to initialize, which delays the emission of metrics.	(not applicable)	Check back after 10 or 20 minutes.

For more information about monitoring metrics and using alarms, see Monitoring Overview. For information about notifications for alarms, see Notifications Overview.

To view default metric charts for resources related to a Compute instance

- **For an attached block volume:** While viewing the instance's details, click **Attached Block Volumes**, and then click the volume you're interested in. Click **Metrics** to see the volume's charts. For more information about the emitted metrics, see Block Volume Metrics.
- **For the attached boot volume:** While viewing the instance's details, click **Boot Volume**, and then click the volume you're interested in. Click **Metrics** to see the volume's charts. For more information about the emitted metrics, see Block Volume Metrics.
- **For an attached VNIC:** While viewing the instance's details, click **Attached VNICs**, and then click the VNIC you're interested in. Click **Metrics** to see the charts for the VNIC. For more information about the emitted metrics, see Networking Metrics.

To view default metric charts for multiple Compute instances

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Service Metrics**.
2. For **Metric Namespace**, select **oci_computeagent**.
   The **Service Metrics** page dynamically updates the page to show charts for each metric that is emitted by the selected metric namespace. For more information about the emitted metrics, see Compute Metrics.

For more information about monitoring metrics and using alarms, see Monitoring Overview. For information about notifications for alarms, see Notifications Overview.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following APIs for monitoring:

- Monitoring API for metrics and alarms
- Notifications API for notifications (used with alarms)

Compute Performance

The content in the sections below apply to **Category 7** and **Section 3.a** of the Oracle PaaS and IaaS Public Cloud Services Pillar documentation.

Oracle Cloud Infrastructure provides a variety of instance configurations in both bare metal and virtual machine (VM) shapes. Each shape varies on multiple dimensions including memory, CPU cores, network bandwidth, and the option of local NVMe SSD storage found in DenseIO shapes.

Oracle Cloud Infrastructure provides a service-level agreement (SLA) for NVMe performance. Measuring performance is complex and open to variability.

A NVMe drive also has non-uniform drive performance over the period of drive usage. A NVMe drive performs differently when tested brand new compared to when tested in a steady-state after some duration of usage. New drives have not incurred many write/erase cycles and the inline garbage collection has not had a significant impact on IOPS performance. To achieve the goal of reproducibility and reduced variability, our testing focuses on the steady-state duration of the NVMe drive's operation.

Testing Methodology



Warning

Before running any tests, protect your data by making a backup of your data and operating system environment to prevent any data loss. The tests described in this document will overwrite the data on the disk, and cause data corruption.

Summary: To capture the IOPS measure, first provision a shape such as the new BM.DenseIO2.52, and then use the Gartner Cloud Harmony test suite to run tests on an instance running the latest supported Oracle Linux image for each NVMe drive target.

Instructions:

1. Launch an instance based on the latest supported Oracle Linux image and select a shape such as the new BM.DenseIO2.52. For launch instructions, see Creating an Instance.
2. Run the Gartner Cloud Harmony test suite tests on the instance for each NVMe drive target. The following is an example of a command that will work for all shapes and drives on the shape:

```
sudo ./run.sh `ls /dev/nvme[0-9]n1 | sed -e 's/\//\--target=\//'`
--nopurge -noprecondition --fio_direct=1 --fio_size=10g --test=iops
--skip_blocksize=512b --skip_blocksize=8k --skip_blocksize=16k
--skip_blocksize=32k --skip_blocksize=64k --skip_blocksize=128k
--skip_blocksize=1m
```

The SLA for NVMe drive performance is measured against 4k block sizes with 100% random write workload on DenseIO shapes where the drive is in a steady-state of operation.

Performance Benchmarks

The following table lists the minimum IOPS for the specified shape to meet the SLA, given the testing methodology with 4k block sizes for 100% random write tests using the tests described in the previous section.

Shape	Minimum Supported IOPS
VM.DenseIO1.4	200k
VM.DenseIO1.8	250k
VM.DenseIO1.16	400k
BM.DenseIO1.36	2.5MM

Shape	Minimum Supported IOPS
VM.DenseIO2.8	250k
VM.DenseIO2.16	400k
VM.DenseIO2.24	800k
BM.DenseIO2.52	3.0MM

While the NVMe drives are capable of higher IOPS, Oracle Cloud Infrastructure currently guarantee this minimum level of IOPS performance.

Frequently Asked Questions

Q: I suspect a slowdown in my NVMe drive performance. Is there a SLA violation?

A: We test hosts on a regular basis to ensure that are our low-level software updates do not regress performance. In the event you have reproduced the testing methodology and your drive's performance does not meet the terms in the SLA please contact your Oracle sales team.

Q: Why does the testing methodology not represent a diversity of IO workloads such as random reads and writes to reflect real world IO?

A: We focused on reproducibility and we believe the tests provide a significant indicator of overall drive performance.

Q: Will Oracle Cloud Infrastructure change the tests in this document?

A: We will make changes to provide greater customer value through better guarantees and improved reproducibility.

Compute Health Monitoring for Bare Metal Instances

Compute health monitoring for bare metal instances is a feature that provides notifications about hardware issues with your bare metal instances. With the health monitoring feature, you can now easily monitor the health of your bare metal instances' hardware, including their components such as DIMM, CPU, motherboard and NVMe drives. You can use the notifications to identify problems enabling you to proactively redeploy your instances to improve availability.

Health monitoring notifications are sent in emails to the tenant administrator within one business day of the error occurring. This warning allows you to take action before any potential hardware failure and redeploy your instances to healthy hardware to minimize the impact on your applications.

Guidance on Compute Health Monitoring Error Messages

This section contains information about the most common health monitoring error messages and provides troubleshooting suggestions for you to try for your bare metal instance.

Click the error message that matches the information in your notification email to expand the troubleshooting section.

1: A fault in the PCI subsystem has been detected

Details: This error indicates that one or more of the PCI devices in your instance have either failed or are not operating at peak performance.

Troubleshooting steps:

- If you cannot connect to the instance over the network, (see Connecting to an Instance), the NIC may have failed. Use the Console or CLI to stop the instance and then start the instance, see Stopping and Starting an Instance.
  If you're still unable to connect to the instance over the network, you may still be able to connect to it using a console connection. Follow the steps in Connecting to the Serial Console or Connecting to the VNC Console to establish a console connection and then

reboot the instance. If the instance remains inaccessible you will need to replace it using the steps in Moving a Compute Instance.

- An NVMe device may have failed.

  On Linux-based systems, run the command `sudo lsblk` to get a list of the attached NVMe devices.

  On Windows-based systems, open Disk Manager. Check the count of NVMe devices against the expected number of devices in Compute Shapes.

  If you determine that an NVMe device is missing from the list of devices for your instance, we recommend replacing the instance using the steps in Moving a Compute Instance.

2:A fault in the memory subsystem was detected

Details: This error indicates that one or more non-critical errors has been detected on a DIMM in your instance. The instance may have unexpectedly rebooted in the last 72 hours.

Troubleshooting steps:

- If the instance has unexpectedly rebooted in the last 72 hours one or more DIMMs may have been disabled.
  - On Linux-based systems, run the command `awk '$3=="kB" {$2=$2/1024**2;$3="GB";} 1' /proc/meminfo | column -t | grep MemTotal` to check for the total amount of memory in the instance.
  - On Windows-based systems, open Resource Monitor to check the memory for the instance.

  If the total memory in the instance is lower than expected, then one or more DIMMs have failed. If this is impacting your application we recommend replacing the instance using the steps in Moving a Compute Instance.
- If the instance has not unexpectedly rebooted it is at increased risk of doing so. During the next reboot one or more DIMMs may be disabled. We recommend replacing the instance using the steps in Moving a Compute Instance.

3: A fault in the memory subsystem was detected during instance launch or a recent reboot

Details: This error indicates that one or more failed DIMMs were detected in your instance while the instance was being launched or rebooted. Any failed DIMMs have been disabled.

Troubleshooting steps: The total amount of memory in the instance will be lower than expected. If this is impacting your application we recommend replacing the instance using the steps in Moving a Compute Instance.

To check for the amount of memory in the instance:

- On Linux-based systems, run the command `awk '$3=="kB" {$2=$2/1024**2;$3="GB";} 1' /proc/meminfo | column -t | grep MemTotal` to check for the total amount of memory in the instance.
- On Windows-based systems, open Resource Monitor to check the memory for the instance.

The expected values are documented in Compute Shapes.

4: A fault has been detected in one or more CPUs

Details: This error indicates that a processor or one or more cores have failed in your instance. Your instance may be inaccessible or there may be fewer available cores than expected.

Troubleshooting steps:

- If the instance is inaccessible you will need to replace it using the steps in Moving a Compute Instance.
- If your instance is available check for the expected number of cores.
    - On Linux-based systems, run the command `nproc --all` to check for the number of cores.
    - On Windows-based systems, open Resource Monitor to to check for the number of cores.

Compare the core count to the expected values documented in Compute Shapes. If the number of cores is less than expected and this reduction is impacting your application we recommend replacing the instance using the steps in Moving a Compute Instance.

5: A fault in the instance management controller has been detected

Details: This error indicates that a device used to manage your instance may have failed. You may not be able to use the Console, CLI, SDKs , or REST APIs to stop, start, or reboot your instance. This functionality will still be available from within the instance using the standard operating system commands. You also may not be able to create a console connection to your instance. You will still be able to terminate your instance.

Troubleshooting steps: If this loss of control is impacting your application we recommend replacing the instance using the steps in Moving a Compute Instance.

CHAPTER 7 Container Engine for Kubernetes

This chapter explains how to define and create Kubernetes clusters to enable the deployment, scaling, and management of containerized applications.

Overview of Container Engine for Kubernetes

Oracle Cloud Infrastructure Container Engine for Kubernetes is a fully-managed, scalable, and highly available service that you can use to deploy your containerized applications to the cloud. Use Container Engine for Kubernetes (sometimes abbreviated to just OKE) when your development team wants to reliably build, deploy, and manage cloud-native applications. You specify the compute resources that your applications require, and Container Engine for Kubernetes provisions them on Oracle Cloud Infrastructure in an existing OCI tenancy.

Container Engine for Kubernetes uses Kubernetes - the open-source system for automating deployment, scaling, and management of containerized applications across clusters of hosts. Kubernetes groups the containers that make up an application into logical units (called pods) for easy management and discovery. Container Engine for Kubernetes uses versions of Kubernetes certified as conformant by the Cloud Native Computing Foundation (CNCF).

You can access Container Engine for Kubernetes to define and create Kubernetes clusters using the Console and the REST API. You can access the clusters you create using the Kubernetes command line (kubectl), the Kubernetes Dashboard, and the Kubernetes API.

Container Engine for Kubernetes is integrated with Oracle Cloud Infrastructure Identity and Access Management (IAM), which provides easy authentication with native Oracle Cloud Infrastructure identity functionality.

For an introductory tutorial, see Creating a Cluster with Oracle Cloud Infrastructure Container Engine for Kubernetes.

Ways to Access Oracle Cloud Infrastructure

You can access Oracle Cloud Infrastructure using the Console (a browser-based interface) or the REST API. Instructions for the Console and API are included in topics throughout this guide. For a list of available SDKs, see Software Development Kits and Command Line Interface.

To access the Console, you must use a supported browser. Oracle Cloud Infrastructure supports the latest desktop versions of Google Chrome, Microsoft Edge, Internet Explorer 11, Safari, Firefox, and Firefox ESR. Note that private browsing mode is not supported for Firefox, Internet Explorer, or Edge. Mobile browsers are not supported.

For general information about using the API, see REST APIs.

Resource Identifiers

Most types of Oracle Cloud Infrastructure resources have a unique, Oracle-assigned identifier called an Oracle Cloud ID (OCID). For information about the OCID format and other ways to identify your resources, see Resource Identifiers.

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

Note that to perform certain operations on clusters created by Container Engine for Kubernetes, you might require additional permissions granted via a Kubernetes RBAC role or clusterrole. See About Access Control and Container Engine for Kubernetes.

Container Engine for Kubernetes Capabilities and Limits

In each region that is enabled for your tenancy, you can create three clusters (Monthly Universal Credits) or one cluster (Pay-as-You-Go or Promo) by default. Each cluster you create can have a maximum of 1000 nodes. See Service Limits.

Required IAM Service Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

If you're new to policies, see Getting Started with Policies and Common Policies.

For more details about policies for Container Engine for Kubernetes, see:

- Policy Configuration for Cluster Creation and Deployment
- Details for Container Engine for Kubernetes

Preparing for Container Engine for Kubernetes

Before you can use Container Engine for Kubernetes to create a Kubernetes cluster:

- You must have access to an Oracle Cloud Infrastructure tenancy. The tenancy must be subscribed to one or more of the regions in which Container Engine for Kubernetes is available (see Availability by Region Name and Region Code).

- Your tenancy must have sufficient quota on different types of resource (see Service Limits). More specifically:
    - Compute instance quota: To create a Kubernetes cluster, at least one compute instance (node) must be available in the tenancy. For example, to create a highly available cluster spanning three availability domains, three compute instances must be available (one in each availability domain in a region).
    - Block volume quota: If you intend to create Kubernetes persistent volumes, sufficient block volume quota must be available in each availability domain to meet the persistent volume claim. Persistent volume claims must request a minimum of 50 gigabytes. See Creating a Persistent Volume Claim.
    - Load balancer quota: If you intend to create a load balancer to distribute traffic between the nodes running a service in a Kubernetes cluster, sufficient load balancer quota must be available in the region. See Creating Load Balancers to Distribute Traffic Between Cluster Nodes.
- Within your tenancy, there must already be a compartment to contain the necessary network resources (such as a VCN, subnets, internet gateway, route table, security lists). If such a compartment does not exist already, you will have to create it. Note that the network resources can reside in the root compartment. However, if you expect multiple teams to create clusters, best practice is to create a separate compartment for each team.
- Within the compartment, network resources (such as a VCN, subnets, internet gateway, route table, security lists) must be appropriately configured in each region in which you want to create and deploy clusters. For example, to create a highly available cluster spanning three availability domains, the VCN must include three subnets in different availability domains for node pools, and optionally (but usually) one or two further subnets for load balancers. When creating a new cluster, you can have Container Engine for Kubernetes automatically create and configure new network resources for the new cluster, or you can specify existing network resources. If you specify existing network resources, you or somebody else must have already configured those resources appropriately. See Network Resource Configuration for Cluster Creation and Deployment.

- Within the root compartment of your tenancy, a policy statement (`Allow service OKE to manage all-resources in tenancy`) must be defined to give Container Engine for Kubernetes access to resources in the tenancy. See Create Policy for Container Engine for Kubernetes (Required)
- To create and/or manage clusters, you must belong to one of the following:
    - The tenancy's Administrators group
    - A group to which a policy grants the appropriate Container Engine for Kubernetes permissions. If you are creating or modifying clusters using the Console, or want Container Engine for Kubernetes to automatically create and configure new network resources for a new cluster, policies must also grant the group the following permissions:
        - VCN_READ and VCN_CREATE
        - SUBNET_READ and SUBNET_CREATE
        - COMPARTMENT_INSPECT
        - INTERNET_GATEWAY_CREATE
        - NAT_GATEWAY_CREATE
        - ROUTE_TABLE_UPDATE
        - SECURITY_LIST_CREATE

      See Create One or More Policies for Groups (Optional).
- To perform operations on a cluster:
    - You must have installed and configured the Kubernetes command line tool kubectl (see the kubectl documentation).
    - You must have downloaded the cluster's kubeconfig configuration file (see Downloading a kubeconfig File to Enable Cluster Access).
    - You must have appropriate permissions to access the cluster (see About Access Control and Container Engine for Kubernetes).

Availability by Region Name and Region Code

Container Engine for Kubernetes is available in the following regions. Note that you have to use the region code in some commands. In some cases, you might have to use shortened versions of availability domain names.

Region Name	Region Code	Shortened Availability Domain Names
Ashburn	iad	• US-ASHBURN-AD-1 • US-ASHBURN-AD-2 • US-ASHBURN-AD-3
Frankfurt	fra	• EU-FRANKFURT-1-AD-1 • EU-FRANKFURT-1-AD-2 • EU-FRANKFURT-1-AD-3
London	lhr	• UK-LONDON-1-AD-1 • UK-LONDON-1-AD-2 • UK-LONDON-1-AD-3
Phoenix	phx	• PHX-AD-1 • PHX-AD-2 • PHX-AD-3
Seoul	icn	• AP-SEOUL-1-AD-1
Tokyo	nrt	• AP-TOKYO-1-AD-1
Toronto	yyz	• CA-TORONTO-1-AD-1

Network Resource Configuration for Cluster Creation and Deployment

Before you can use Container Engine for Kubernetes to create and deploy clusters in the regions in a tenancy:

- The tenancy's root compartment must include a policy to allow Container Engine for Kubernetes to perform operations in the tenancy. See Create Policy for Container Engine for Kubernetes (Required).
- Within the tenancy, there must already be a compartment to contain the necessary network resources (such as a VCN, subnets, internet gateway, route table, security lists). If such a compartment does not exist already, you will have to create it. Note that the network resources can reside in the root compartment. However, if you expect multiple teams to create clusters, best practice is to create a separate compartment for each team.
- Within the compartment, network resources (such as a VCN, subnets, internet gateway, route table, security lists) must be appropriately configured in each region in which you want to create and deploy clusters. When creating a new cluster, you can have Container Engine for Kubernetes automatically create and configure new network resources for a new 'quick cluster'. Alternatively, you can explicitly specify the existing network resources to use for a 'custom cluster'. If you specify existing network resources, you or somebody else must have already configured those resources appropriately, as described in this topic.

This topic describes the necessary configuration for each network resource. To see details of a typical configuration, see Example Network Resource Configurations.

For an introductory tutorial, see Creating a Cluster with Oracle Cloud Infrastructure Container Engine for Kubernetes.

Root Compartment Configuration

You have to define a policy for the tenancy's root compartment to enable Container Engine for Kubernetes to perform operations on the tenancy. See Create Policy for Container Engine for Kubernetes (Required).

VCN Configuration

The VCN in which you want to create and deploy clusters must be configured as follows:

- The VCN must have a CIDR block defined that is large enough for the number of subnets you specify for the clusters you create. For example, to create a highly available cluster will typically require five subnets to support the necessary number of hosts and load balancers. However, you can create clusters with fewer subnets. A /16 CIDR block would be large enough for almost all use cases (10.0.0.0/16 for example). The CIDR block you specify for the VCN must not overlap with the CIDR block you specify for pods and for the Kubernetes services (see CIDR Blocks and Container Engine for Kubernetes).
- The VCN must have an internet gateway defined. See Internet Gateway Configuration.
- If you intend to deploy worker nodes in private subnets, the VCN must have a NAT gateway defined. See NAT Gateway Configuration.
- The VCN must have a route table defined that has a route rule specifying the internet gateway as the target for the destination CIDR block. If you intend to deploy worker nodes in private subnets, the route table must also have a route rule specifying the NAT gateway as the target for the destination CIDR block. See Route Table Configuration.
- The VCN must have an appropriate number of subnets defined. For example, to create a typical highly available cluster will require three subnets in different availability domains in which to deploy worker nodes, and two subnets to host load balancers. However, you can create clusters with fewer worker nodes, and fewer or no load balancers, and therefore require fewer subnets. See Subnet Configuration.
- The VCN must have security lists defined for worker node subnets and load balancer subnets (if specified). See Security List Configuration.

In addition, Oracle recommends DNS Resolution is selected for the VCN.

See VCNs and Subnets and Example Network Resource Configurations.

Internet Gateway Configuration

The VCN in which you want to create and deploy clusters must have an internet gateway. The internet gateway must be specified as the target for the destination CIDR block 0.0.0.0/0 in a

route rule in a route table.

See VCNs and Subnets and Example Network Resource Configurations.

NAT Gateway Configuration

If you intend to deploy worker nodes in private subnets, the VCN must have a NAT gateway (in addition to an internet gateway) to enable the worker nodes to initiate connections to the internet. The NAT gateway must be specified as the target for the destination CIDR block 0.0.0.0/0 in a route rule in a separate route table.

See NAT Gateway and Example Network Resource Configurations.

Route Table Configuration

The VCN in which you want to create and deploy clusters must have a route table. The route table must have a route rule that specifies an internet gateway as the target for the destination CIDR block 0.0.0.0/0.

If you intend to deploy worker nodes in private subnets, a separate route table must have a route rule that specifies a NAT gateway as the target for the destination CIDR block 0.0.0.0/0 to enable the worker nodes to initiate connections to the internet. This route table is in addition to the route table specifying an internet gateway as a target.

See Internet Gateway and Example Network Resource Configurations.

DHCP Options Configuration

The VCN in which you want to create and deploy clusters must have DHCP Options configured. The default value for **DNS Type** of Internet and VCN Resolver is acceptable.

See DHCP Options and Example Network Resource Configurations.

Security List Configuration

The VCN in which you want to create and deploy clusters must have security lists defined for worker node subnets and load balancer subnets (if specified). The security lists for worker

node subnets and load balancer subnets must be different. The security list for load balancer subnets must be unique and for their exclusive use.

Worker nodes are created with public or private IP addresses, according to whether you specify public or private subnets when defining the node pools in a cluster. In both cases, worker nodes must be able to make outbound connections to the internet. Container Engine for Kubernetes must also be able to access worker nodes.

See Security Lists and Example Network Resource Configurations.

Public Worker Node Subnet Security List configuration

When configuring a security list for public worker node subnets, the security list must have:

- stateless ingress and egress rules that allow all traffic between the different worker node subnets
- stateless ingress and egress rules that allow all traffic between worker node subnets and load balancer subnets (if specified)
- an egress rule that allows all outbound traffic to the internet
- ingress rules to allow Container Engine for Kubernetes to access worker nodes on port 22 from the following source CIDR blocks:
    - 130.35.0.0/16
    - 134.70.0.0/17
    - 138.1.0.0/16
    - 140.91.0.0/17
    - 147.154.0.0/16
    - 192.29.0.0/16

Optionally, you can include ingress rules for public worker node subnets to:

- explicitly allow SSH access to worker nodes on port 22 (see Connecting to Worker Nodes in Public Subnets Using SSH)
- allow inbound traffic to the worker nodes on the default NodePort range of 30000 to 32767 (see the Kubernetes documentation)

Private Worker Node Subnet Security List configuration

When configuring a security list for private worker node subnets, the security list must have:

- stateless ingress and egress rules that allow all traffic between the different worker node subnets
- stateless ingress and egress rules that allow all traffic between worker node subnets and load balancer subnets
- an egress rule that allows all outbound traffic to the internet

Optionally, you can include ingress rules for private worker node subnets to:

- explicitly allow SSH access to worker nodes on port 22 from within the VCN CIDR block (see Connecting to Worker Nodes in Private Subnets Using SSH)
- allow inbound traffic to the worker nodes on the default NodePort range of 30000 to 32767 from within the VCN CIDR block (see the Kubernetes documentation)

Subnet Configuration

The characteristics of the cluster you want to create, and the number of availability domains in the region in which you want to deploy the cluster, will determine the number of subnets to configure. For example, to create a typical highly available cluster will require three subnets in different availability domains in which to deploy worker nodes, and two subnets to host load balancers.



Note

Note that Container Engine for Kubernetes does not yet support regional subnets, so you cannot currently deploy worker nodes or host load balancers in regional subnets.

The VCN in which you want to create and deploy clusters must have at least one subnet defined in which to deploy worker nodes. Worker node subnets can be either public, or private

for additional security (as specified by the **Subnet access** property). The number of worker node subnets to create depends on the region in which you are creating the cluster:

- If you are creating a cluster in a region with multiple availability domains, you can define multiple worker node subnets. Each worker node subnet must be in a different availability domain.
- If you are creating a cluster in a region with a single availability domain, you can only define one worker node subnet.

You have the option to define and use load balancers in clusters you create. If you want to define and use load balancers, the VCN in which you want to create and deploy clusters must have at least one subnet defined to host the load balancers. Load balancer subnets can be public or private (as specified by the **Subnet access** property) and must each be in a separate availability domain. However, load balancers are optional, so you might not define load balancer subnets at all. The number of load balancer subnets to define depends on the region in which you are creating the cluster:

- If you are creating a cluster in a region with three availability domains, you can define zero or two load balancer subnets. If you define two load balancer subnets, the two load balancer subnets must be in different availability domains.
- If you are creating a cluster in a region with a single availability domain, you can define zero or one load balancer subnet.

In addition, all subnets must have the following properties set as shown:

- **Route Table:** The name of the route table that has a route rule specifying an internet gateway (for public worker node subnets) or NAT gateway (for private worker node subnets) as the target for the destination CIDR block 0.0.0.0/0
- **DHCP options:** Default

The CIDR blocks you specify for worker node and load balancer subnets must not overlap with CIDR blocks you specify for pods running in the cluster (see CIDR Blocks and Container Engine for Kubernetes).

Worker node subnets must have different security lists to load balancer subnets.

See VCNs and Subnets and Example Network Resource Configurations.

Example Network Resource Configurations

Before you can use Container Engine for Kubernetes to create and deploy clusters in the regions in a tenancy:

- The tenancy's root compartment must include a policy to allow Container Engine for Kubernetes to perform operations in the tenancy. See Create Policy for Container Engine for Kubernetes (Required).
- Within the tenancy, there must already be a compartment to contain the necessary network resources (such as a VCN, subnets, internet gateway, route table, security lists). If such a compartment does not exist already, you will have to create it. Note that the network resources can reside in the root compartment. However, if you expect multiple teams to create clusters, best practice is to create a separate compartment for each team.
- Within the compartment, network resources (such as a VCN, subnets, internet gateway, route table, security lists) must be appropriately configured in each region in which you want to create and deploy clusters. When creating a new cluster, you can have Container Engine for Kubernetes automatically create and configure new network resources for a new 'quick cluster'. Alternatively, you can explicitly specify the existing network resources to use for a 'custom cluster'. If you specify existing network resources, you or somebody else must have already configured those resources appropriately. See Network Resource Configuration for Cluster Creation and Deployment.

This topic gives examples of how you might configure network resources for highly available cluster creation and deployment in a region with three availability domains:

- for public clusters, where you want worker nodes hosted in public subnets that can be accessed directly from the internet (see Example 1: Example Network Resource Configuration for a Highly Available Public Cluster)
- for private clusters, where you want worker nodes hosted in private subnets that can only be accessed from within the VCN (see Example 2: Example Network Resource Configuration for a Highly Available Private Cluster)

For an introductory tutorial, see Creating a Cluster with Oracle Cloud Infrastructure Container Engine for Kubernetes.

Example 1: Example Network Resource Configuration for a Highly Available Public Cluster

This example assumes you want worker nodes hosted in three public subnets that can be accessed directly from the internet.

EXAMPLE NETWORK RESOURCE CONFIGURATION

Resource	Example
VCN	Created manually, and defined as follows: • **Name:** acme-dev-vcn • **CIDR Block:** 10.0.0.0/16 • **DNS Resolution:** Selected
Internet Gateway	Created manually, and defined as follows: • **Name:** gateway-0
Route Table	Created manually, and defined as follows: • **Name:** routetable-0 A route rule defined as follows: • **Destination CIDR block:** 0.0.0.0/0 • **Target Type:** Internet Gateway • **Target Internet Gateway:** gateway-0
DHCP Options	Created automatically and defined as follows: • **DNS Type** set to Internet and VCN Resolver

Resource	Example
Security Lists	Two created (in addition to the default security list) manually, named, and defined as follows: • **Security List Name:** workers • **Security List Name:** loadbalancers For details of the ingress rules and egress rules defined for the workers security list and the loadbalancers security list, see Example Security List Configurations for a Highly Available Public Cluster.

Resource	Example
Subnets	Three worker subnets created manually, named, and defined as follows: • **Name:** workers-1 with the following properties: ◦ **Availability Domain:** AD1 ◦ **CIDR Block:** 10.0.10.0/24 ◦ **Route Table:** routetable-0 ◦ **Subnet access:** Public ◦ **DNS Resolution:** Selected ◦ **DHCP Options:** Default ◦ **Security List:** workers • **Name:** workers-2 with the following properties: ◦ **Availability Domain:** AD2 ◦ **CIDR Block:** 10.0.11.0/24 ◦ **Route Table:** routetable-0 ◦ **Subnet access:** Public ◦ **DNS Resolution:** Selected ◦ **DHCP Options:** Default ◦ **Security List:** workers • **Name:** workers-3 with the following properties: ◦ **Availability Domain:** AD3 ◦ **CIDR Block:** 10.0.12.0/24 ◦ **Route Table:** routetable-0 ◦ **Subnet access:** Public ◦ **DNS Resolution:** Selected ◦ **DHCP Options:** Default

Resource	Example
	◦ **Security List:** workers Two load balancer subnets created, named, and defined as follows: • **Name:** loadbalancers-1 with the following properties: ◦ **Availability Domain:** AD1 ◦ **CIDR Block:** 10.0.20.0/24 ◦ **Route Table:** routetable-0 ◦ **Subnet access:** Public ◦ **DNS Resolution:** Selected ◦ **DHCP Options:** Default ◦ **Security List:** loadbalancers • **Name:** loadbalancers-2 with the following properties: ◦ **Availability Domain:** AD2 ◦ **CIDR Block:** 10.0.21.0/24 ◦ **Route Table:** routetable-0 ◦ **Subnet access:** Public ◦ **DNS Resolution:** Selected ◦ **DHCP Options:** Default ◦ **Security List:** loadbalancers

EXAMPLE SECURITY LIST CONFIGURATIONS FOR A HIGHLY AVAILABLE PUBLIC CLUSTER

In the example VCN, two security lists have been created (in addition to the default security list) to control access to and from the public worker node subnets and load balancer subnets. The two security lists are named 'workers' and 'loadbalancers' respectively.

The workers security list has the following ingress and egress rules for public worker node subnets:

Example Ingress Rules in a Security List for a Public Worker Node Subnet:

#	Type	Source CIDR	IP Protocol	Source Port Range	Dest. Port Range	Type and Code	Allows: and Description:
1	Stateless	10.0.10.0/24	All	n/a	n/a	n/a	**Allows:** All traffic for all ports **Description:** This rule enables intra-VCN traffic.
2	Stateless	10.0.11.0/24	All	n/a	n/a	n/a	**Allows:** All traffic for all ports **Description:** This rule enables intra-VCN traffic.
3	Stateless	10.0.12.0/24	All	n/a	n/a	n/a	**Allows:** All traffic for all ports **Description:** This rule enables intra-VCN traffic.

#	Type	Source CIDR	IP Protocol	Source Port Range	Dest. Port Range	Type and Code	Allows: and Description:
4	Stateful	0.0.0.0/0	ICMP	n/a	n/a	3, 4	**Allows:** ICMP traffic for: 3, 4 Destination Unreachable: Fragmentation Needed and Don't Fragment was Set **Description:** This rule enables worker nodes to receive Path MTU Discovery fragmentation messages.
5	Stateful	130.35.0.0/16	TCP	All	22	n/a	**Allows:** TCP traffic for ports: 22 SSH Remote Login Protocol **Description:** This rule enables Container Engine for Kubernetes to access worker nodes.

#	Type	Source CIDR	IP Protocol	Source Port Range	Dest. Port Range	Type and Code	Allows: and Description:
6	Stateful	134.70.0.0/17	TCP	All	22	n/a	**Allows:** TCP traffic for ports: 22 SSH Remote Login Protocol **Description:** This rule enables Container Engine for Kubernetes to access worker nodes.
7	Stateful	138.1.0.0/16	TCP	All	22	n/a	**Allows:** TCP traffic for ports: 22 SSH Remote Login Protocol **Description:** This rule enables Container Engine for Kubernetes to access worker nodes.

#	Type	Source CIDR	IP Protocol	Source Port Range	Dest. Port Range	Type and Code	Allows: and Description:
8	Stateful	140.91.0.0/17	TCP	All	22	n/a	**Allows:** TCP traffic for ports: 22 SSH Remote Login Protocol **Description:** This rule enables Container Engine for Kubernetes to access worker nodes.
9	Stateful	147.154.0.0/16	TCP	All	22	n/a	**Allows:** TCP traffic for ports: 22 SSH Remote Login Protocol **Description:** This rule enables Container Engine for Kubernetes to access worker nodes.

#	Type	Source CIDR	IP Protocol	Source Port Range	Dest. Port Range	Type and Code	Allows: and Description:
10	Stateful	192.29.0.0/16	TCP	All	22	n/a	**Allows:** TCP traffic for ports: 22 SSH Remote Login Protocol **Description:** This rule enables Container Engine for Kubernetes to access worker nodes.
11	Stateful	0.0.0.0/0	TCP	All	22	n/a	**Allows:** TCP traffic for ports: 22 SSH Remote Login Protocol **Description:** This *optional* rule enables inbound SSH traffic from the internet on port 22 to access worker nodes.

#	Type	Source CIDR	IP Protocol	Source Port Range	Dest. Port Range	Type and Code	Allows: and Description:
12	Stateful	0.0.0.0/0	TCP	All	30000 - 32767	n/a	**Allows:** TCP traffic for ports: 30000 - 32767 **Description:** This *optional* rule enables inbound traffic to the worker nodes on the default NodePort range of 30000-32767 (see the Kubernetes documentation).

Example Egress Rules in a Security List for a Public Worker Node Subnet:

#	Type	Dest. CIDR	IP Protocol	Source Port Range	Dest. Port Range	Type and Code	Allows: and Description:
1	Stateless	10.0.10.0/24	All	n/a	n/a	n/a	**Allows:** All traffic for all ports **Description:** This rule enables intra-VCN traffic.
2	Stateless	10.0.11.0/24	All	n/a	n/a	n/a	**Allows:** All traffic for all ports **Description:** This rule enables intra-VCN traffic.
3	Stateless	10.0.12.0/24	All	n/a	n/a	n/a	**Allows:** All traffic for all ports **Description:** This rule enables intra-VCN traffic.
4	Stateful	0.0.0.0/0	All	n/a	n/a	n/a	**Allows:** All traffic for all ports **Description:** This rule enables outbound access to the internet.

The loadbalancers security list has the following ingress and egress rules for load balancer subnets:

Example Ingress Rules in a Security List for a Load Balancer Subnet:

#	Type	Source CIDR	IP Protocol	Source Port Range	Dest. Port Range	Type and Code	Allows: and Description:
1	Stateless	0.0.0.0/0	TCP	All	All	n/a	**Allows:** TCP traffic for all ports: all **Description:** This rule enables incoming public traffic to service load balancers.

Example Egress Rules in a Security List for a Load Balancer Subnet:

#	Type	Dest. CIDR	IP Protocol	Source Port Range	Dest. Port Range	Type and Code	Allows: and Description:
1	Stateless	0.0.0.0/0	TCP	All	All	n/a	**Allows:** TCP traffic for ports: all **Description:** This rule enables responses from a web application through the service load balancers.

Example 2: Example Network Resource Configuration for a Highly Available Private Cluster

This example assumes you want worker nodes hosted in three private subnets that can only be accessed from within the VCN.

EXAMPLE NETWORK RESOURCE CONFIGURATION

Resource	Example
VCN	Created manually, and defined as follows: • **Name:** acme-dev-vcn • **CIDR Block:** 10.0.0.0/16 • **DNS Resolution:** Selected
Internet Gateway	Created manually, and defined as follows: • **Name:** gateway-0
NAT Gateway	Created manually, and defined as follows: • **Name:** nat-gateway-0
Route Table	Two route tables created manually, named, and defined as follows: • **Name:** routetable-0, with a route rule defined as follows: ◦ **Destination CIDR block:** 0.0.0.0/0 ◦ **Target Type:** Internet Gateway ◦ **Target Internet Gateway:** gateway-0 • **Name:** routetable-1, with a route rule defined as follows: ◦ **Destination CIDR block:** 0.0.0.0/0 ◦ **Target Type:** NAT Gateway ◦ **Target NAT Gateway:** nat-gateway-0

Resource	Example
DHCP Options	Created automatically and defined as follows: • **DNS Type** set to Internet and VCN Resolver
Security Lists	Two created (in addition to the default security list) manually, named, and defined as follows: • **Security List Name:** workers • **Security List Name:** loadbalancers For details of the ingress rules and egress rules defined for these security lists, see Example Security List Configurations for a Highly Available Private Cluster.

Resource	Example
Subnets	Three worker subnets created manually, named, and defined as follows: • **Name:** workers-1 with the following properties: ◦ **Availability Domain:** AD1 ◦ **CIDR Block:** 10.0.10.0/24 ◦ **Route Table:** routetable-1 ◦ **Subnet access:** Private ◦ **DNS Resolution:** Selected ◦ **DHCP Options:** Default ◦ **Security List:** workers • **Name:** workers-2 with the following properties: ◦ **Availability Domain:** AD2 ◦ **CIDR Block:** 10.0.11.0/24 ◦ **Route Table:** routetable-1 ◦ **Subnet access:** Private ◦ **DNS Resolution:** Selected ◦ **DHCP Options:** Default ◦ **Security List:** workers • **Name:** workers-3 with the following properties: ◦ **Availability Domain:** AD3 ◦ **CIDR Block:** 10.0.12.0/24 ◦ **Route Table:** routetable-1 ◦ **Subnet access:** Private ◦ **DNS Resolution:** Selected ◦ **DHCP Options:** Default

Resource	Example
	◦ **Security List:** workers Two load balancer subnets created, named, and defined as follows: • **Name:** loadbalancers-1 with the following properties: ◦ **Availability Domain:** AD1 ◦ **CIDR Block:** 10.0.20.0/24 ◦ **Route Table:** routetable-0 ◦ **Subnet access:** Public ◦ **DNS Resolution:** Selected ◦ **DHCP Options:** Default ◦ **Security List:** loadbalancers • **Name:** loadbalancers-2 with the following properties: ◦ **Availability Domain:** AD2 ◦ **CIDR Block:** 10.0.21.0/24 ◦ **Route Table:** routetable-0 ◦ **Subnet access:** Public ◦ **DNS Resolution:** Selected ◦ **DHCP Options:** Default ◦ **Security List:** loadbalancers

Example Security List Configurations for a Highly Available Private Cluster

In the example VCN, two security lists have been created (in addition to the default security list) to control access to and from the private worker node subnets and load balancer subnets. The two security lists are named 'workers' and 'loadbalancers' respectively.

The workers security list has the following ingress and egress rules for private worker node subnets:

Example Ingress Rules in a Security List for a Private Worker Node Subnet:

#	Type	Source CIDR	IP Protocol	Source Port Range	Dest. Port Range	Type and Code	Allows: and Description:
1	Stateless	10.0.10.0/24	All	n/a	n/a	n/a	**Allows:** All traffic for all ports **Description:** This rule enables intra-VCN traffic.
2	Stateless	10.0.11.0/24	All	n/a	n/a	n/a	**Allows:** All traffic for all ports **Description:** This rule enables intra-VCN traffic.

#	Type	Source CIDR	IP Protocol	Source Port Range	Dest. Port Range	Type and Code	Allows: and Description:
3	Stateless	10.0.12.0/24	All	n/a	n/a	n/a	**Allows:** All traffic for all ports **Description:** This rule enables intra-VCN traffic.
4	Stateful	10.0.0.0/16	TCP	All	22	n/a	**Allows:** TCP traffic for ports: 22 SSH Remote Login Protocol **Description:** This *optional* rule enables inbound SSH traffic from the VCN on port 22 to access worker nodes.

Example Egress Rules in a Security List for a Private Worker Node Subnet:

#	Type	Dest. CIDR	IP Protocol	Source Port Range	Dest. Port Range	Type and Code	Allows: and Description:
1	Stateless	10.0.10.0/24	All	n/a	n/a	n/a	**Allows:** All traffic for all ports **Description:** This rule enables intra-VCN traffic.
2	Stateless	10.0.11.0/24	All	n/a	n/a	n/a	**Allows:** All traffic for all ports **Description:** This rule enables intra-VCN traffic.
3	Stateless	10.0.12.0/24	All	n/a	n/a	n/a	**Allows:** All traffic for all ports **Description:** This rule enables intra-VCN traffic.
4	Stateful	0.0.0.0/0	All	n/a	n/a	n/a	**Allows:** All traffic for all ports **Description:** This rule enables outbound access to the internet.

The loadbalancers security list has the following ingress and egress rules for load balancer subnets:

Example Ingress Rules in a Security List for a Load Balancer Subnet:

#	Type	Source CIDR	IP Protocol	Source Port Range	Dest. Port Range	Type and Code	Allows: and Description:
1	Stateless	0.0.0.0/0	TCP	All	All	n/a	**Allows:** TCP traffic for all ports: all **Description:** This rule enables incoming public traffic to service load balancers.

Example Egress Rules in a Security List for a Load Balancer Subnet:

#	Type	Dest. CIDR	IP Protocol	Source Port Range	Dest. Port Range	Type and Code	Allows: and Description:
1	Stateless	0.0.0.0/0	TCP	All	All	n/a	**Allows:** TCP traffic for ports: all **Description:** This rule enables responses from a web application through the service load balancers.

CIDR Blocks and Container Engine for Kubernetes

When configuring the VCN and the worker node and load balancer subnets for use with Container Engine for Kubernetes, you specify CIDR blocks to indicate the network addresses that can be allocated to the resources. See Network Resource Configuration for Cluster Creation and Deployment.

When creating a cluster with Container Engine for Kubernetes, you specify:

- CIDR blocks for the Kubernetes services
- CIDR blocks that can be allocated to pods running in the cluster (see Creating a Kubernetes Cluster)

Note the following:

- The CIDR block you specify for the VCN must not overlap with the CIDR block you specify for the Kubernetes services.
- The CIDR blocks you specify for pods running in the cluster must not overlap with CIDR blocks you specify for worker node and load balancer subnets.

Policy Configuration for Cluster Creation and Deployment

Before you can use Container Engine for Kubernetes to create and deploy clusters in the regions in a tenancy, the tenancy's root compartment must include a policy to allow Container Engine for Kubernetes to perform operations in the tenancy. See Create Policy for Container Engine for Kubernetes (Required).

When a tenancy is created, an Administrators group is automatically created for the tenancy. Users that are members of the Administrators group can perform any operation on resources in the tenancy. If all the users that will be working with Container Engine for Kubernetes are already members of the Administrators group, there's no need to create additional policies. However, if you want to enable users that are not members of the Administrators group to use Container Engine for Kubernetes, you must write policies to enable the groups to which those users do belong to perform operations on cluster-related resources in the tenancy. See Create One or More Policies for Groups (Optional).

Note that in addition to the above policies managed by IAM, you can also use the Kubernetes RBAC Authorizer to enforce additional fine-grained access control for users on specific clusters via Kubernetes RBAC roles and clusterroles. See About Access Control and Container Engine for Kubernetes.

Create Policy for Container Engine for Kubernetes (Required)

To create and manage clusters in your tenancy, Container Engine for Kubernetes must have access to all resources in the tenancy. To give Container Engine for Kubernetes the necessary access, create a policy for the service as follows:

1. In the Console, open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Policies**. A list of the policies in the compartment you're viewing is displayed.
2. Select the tenancy's root compartment from the list on the left.
3. Click **Create Policy**.
4. Enter the following:
    - **Name:** A unique name for the policy (for example, `oke-service`). The name must be unique across all policies in your tenancy. You cannot change this later. Avoid entering confidential information.
    - **Description:** A friendly description. You can change this later if you want to. Avoid entering confidential information.
    - **Policy Versioning:** Select **Keep Policy Current** if you'd like the policy to stay current with any future changes to the service's definitions of verbs and resources. Or if you'd prefer to limit access according to the definitions that were current on a specific date, select **Use Version Date** and enter that date in YYYY-MM-DD format. For more information, see Policy Language Version.
    - **Statement:** The following policy statement:

      ```
      Allow service OKE to manage all-resources in tenancy
      ```

    - **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To

apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

5. Click **Create**.

Create One or More Policies for Groups (Optional)

To enable users that are not members of the Administrators group to use Container Engine for Kubernetes, create policies to enable the groups to which those users do belong to perform operations on cluster-related resources in the tenancy as follows:

1. In the Console, open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Policies**. A list of the policies in the compartment you're viewing is displayed.
2. Select the tenancy's root compartment from the list on the left.
3. Click **Create Policy**.
4. Enter the following:
   - **Name:** A unique name for the policy (for example, `acme-dev-team-oke-policy`). The name must be unique across all policies in your tenancy. You cannot change this later. Avoid entering confidential information.
   - **Description:** A friendly description. You can change this later if you want to. Avoid entering confidential information.
   - **Policy Versioning:** Select **Keep Policy Current** if you'd like the policy to stay current with any future changes to the service's definitions of verbs and resources. Or if you'd prefer to limit access according to the definitions that were current on a specific date, select **Use Version Date** and enter that date in YYYY-MM-DD format. For more information, see Policy Language Version.

- **Statement:** A suitable policy statement to allow existing groups to perform operations on cluster-related resources in the tenancy. For example:
  - To enable users in the acme-dev-team group to perform any operation on cluster-related resources, enter a policy statement like `Allow group acme-dev-team to manage cluster-family in tenancy`. This 'catch-all' policy effectively makes all users administrators insofar as cluster-related resources are concerned.
  - To enable users in the acme-dev-team group to create or modify clusters using the Console, or to automatically create and configure associated new network resources when creating new 'quick clusters', policies must also grant the group:
    - the VCN_READ and VCN_CREATE permissions. Enter a policy statement like `Allow group acme-dev-team to manage vcns in tenancy`
    - the SUBNET_READ and SUBNET_CREATE permissions. Enter a policy statement like `Allow group acme-dev-team to manage subnets in tenancy`
    - the COMPARTMENT_INSPECT permission. Enter a policy statement like `Allow group acme-dev-team to inspect compartments in tenancy`
    - the INTERNET_GATEWAY_CREATE permission. Enter a policy statement like `Allow group acme-dev-team to manage internet-gateways in tenancy`
    - the NAT_GATEWAY_CREATE permission. Enter a policy statement like `Allow group acme-dev-team to manage nat-gateways in tenancy`
    - the ROUTE_TABLE_UPDATE permission. Enter a policy statement like `Allow group acme-dev-team to manage route-tables in tenancy`

        - the SECURITY_LIST_CREATE permission. Enter a policy statement like `Allow group acme-dev-team to manage security-lists in tenancy`
    - To enable users in the acme-dev-team-cluster-viewers group to simply list the clusters, enter a policy statement like `Allow group acme-dev-team-cluster-viewers to inspect clusters in tenancy`.
    - To enable users in the acme-dev-team-pool-admins group to list, create, update, and delete node pools, enter a policy statement like `Allow group acme-dev-team-pool-admins to use cluster-node-pools in tenancy`.
    - To enable users in the acme-dev-team-auditors group to see details of operations performed on clusters, enter a policy statement like `Allow group acme-dev-team-auditors to read cluster-work-requests in tenancy`.
- **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

5. Click **Create**.

About Kubernetes Clusters and Nodes

A Kubernetes cluster is a group of nodes. The nodes are the machines running applications. Each node can be a physical machine or a virtual machine. The node's capacity (its number of CPUs and amount of memory) is defined when the node is created. A cluster can be organized into namespaces, to divide the cluster's resources between multiple uses. A cluster comprises:

- one or more master nodes (for high availability, typically there will be a number of master nodes)
- one or more worker nodes (sometimes known as minions)

The master nodes in a cluster run a number of processes:

- kube-apiserver to support API operations via the Kubernetes command line tool (kubectl) and the REST API, and includes admissions controllers required for advanced Kubernetes operations
- kube-controller-manager to manage different Kubernetes components (for example, replication controller, endpoints controller, namespace controller, and serviceaccounts controller)
- kube-scheduler to control where in the cluster to run jobs
- etcd to store the cluster's configuration data

Each worker node runs two Kubernetes processes:

- kubelet to communicate with the master nodes
- kube-proxy to handle networking

Each worker node also runs the Docker runtime.

The Kubernetes processes running on the master nodes are collectively referred to as the Kubernetes Control Plane. Together, the Control Plane processes monitor and record the state of the cluster and distribute requested operations between the nodes in the cluster.

Where an application running on a worker node comprises multiple containers, Kubernetes groups the containers into a single logical unit called a pod for easy management and discovery. The containers in the pod share the same networking namespace and the same storage space, and can be managed as a single object by the Kubernetes Control Plane. A number of pods providing the same functionality can be grouped into a single logical set known as a service.

A Kubernetes manifest file comprises instructions in a yaml or json file that specify how to deploy an application to the node or nodes in a Kubernetes cluster. The instructions include information about the Kubernetes deployment, the Kubernetes service, and other Kubernetes objects to be created on the cluster. The manifest is commonly also referred to as a pod spec, or as a deployment.yaml file (although other filenames are allowed). The parameters to include in a Kubernetes manifest file are described in the Kubernetes documentation.

A node pool is a subset of machines within a cluster that all have the same configuration. Node pools enable you to create pools of machines within a cluster that have different configurations. For example, you might create one pool of nodes in a cluster as virtual machines, and another pool of nodes as bare metal machines. A cluster must have a minimum of one node pool, but a node pool need not contain any worker nodes.

Creating a Kubernetes Cluster

You can use Container Engine for Kubernetes to create new Kubernetes clusters. To create a cluster, you must either belong to the tenancy's Administrators group, or belong to a group to which a policy grants the CLUSTER_MANAGE permission.

You first specify basic details for the new cluster (the cluster name, and the Kubernetes version to install on master nodes). You can then create the cluster in one of two ways:

- Using default settings to create a 'quick cluster' with new network resources as required. This approach is the fastest way to create a new cluster. If you accept all the default values, you can create a new cluster in just a few clicks. New network resources for the cluster are created automatically, along with a node pool and worker nodes. Note that worker nodes in a 'quick cluster' are created in private subnets, so a NAT gateway is also created (in addition to an internet gateway). To create a 'quick cluster', you must belong to a group to which a policy grants the necessary permissions to create the new network resources (see Create One or More Policies for Groups (Optional)).
- Using custom settings to create a 'custom cluster'. This approach gives you the most control over the new cluster. You can explicitly define the new cluster's properties. And you can explicitly specify which existing network resources to use, including the existing public or private subnets in which to create worker nodes. Note that although you will usually define node pools immediately when defining a new 'custom cluster', you don't have to. You can create a 'custom cluster' with no node pools, and add node pools later.



Note

Note that Container Engine for Kubernetes does not yet support regional subnets, so you cannot currently select a regional subnet when specifying a load balancer subnet or a worker node subnet.

Regardless of how you create a cluster, Container Engine for Kubernetes gives names to worker nodes in the following format:

`oke-c<part-of-cluster-OCID>-n<part-of-node-pool-OCID>-s<part-of-subnet-OCID>-<slot>`

where:

- `oke` is the standard prefix for all worker nodes created by Container Engine for Kubernetes
- `c<part-of-cluster-OCID>` is a portion of the cluster's OCID, prefixed with the letter `c`
- `n<part-of-node-pool-OCID>` is a portion of the node pool's OCID, prefixed with the letter `n`
- `s<part-of-subnet-OCID>` is a portion of the subnet's OCID, prefixed with the letter `s`
- `<slot>` is an ordinal number of the node in the subnet (for example, `0`, `1`)

For example, if you specified a cluster is to have two nodes per subnet in a node pool, the two nodes might be named:

- `oke-cywiqripuyg-nsgagklgnst-st2qczvnmba-0`
- `oke-cywiqripuyg-nsgagklgnst-st2qczvnmba-1`

To ensure high availability, Container Engine for Kubernetes creates the Kubernetes Control Plane on multiple Oracle-managed master nodes (distributed across different availability domains in a region where supported).

Using the Console to create a 'Quick Cluster' with Default Settings

To create a 'quick cluster' with default settings and new network resources using Container Engine for Kubernetes:

1. In the Console, open the navigation menu. Under **Solutions, Platform and Edge**, go to **Developer Services** and click **Container Clusters**.
2. Choose a **Compartment** you have permission to work in, and in which you want to create both the new cluster and the associated network resources.
3. On the **Cluster List** page, click **Create Cluster**.
4. Either just accept the default configuration details for the new cluster, or specify alternatives as follows:
    - **Name:** The name of the new cluster. Either accept the default name or enter a name of your choice. Avoid entering confidential information.
    - **Kubernetes Version:** The version of Kubernetes to run on the master nodes and worker nodes of the cluster. Either accept the default version or select a version of your choice. Amongst other things, the Kubernetes version you select determines the default set of admission controllers that are turned on in the created cluster (the set follows the recommendation given in the Kubernetes documentation for that version).
5. Select **Quick Create** to create a new cluster with default settings, along with new network resources for the new cluster.

    The **Create Virtual Cloud Network** panel shows the network resources that will be created for you by default.

    The **Create Node Pool** panel shows the fixed properties of the first node pool in the cluster that will be created for you:
    - the name of the node pool (always pool1)
    - the compartment in which the node pool will be created (always the same as the one in which the new network resources will reside)

- the version of Kubernetes that will run on each worker node in the node pool (always the same as the version specified for the master nodes)
- the image to use on each node in the node pool

The **Create Node Pool** panel also contains some node pool properties that you can change, but which have been given sensible defaults.

6. Either just accept all the default configuration details and skip ahead to the next step to create the cluster immediately, or specify alternatives as follows:
   a. Either accept the default configuration details for the node pool, or specify alternatives in the **Create Node Pool** panel as follows:
      - **Shape:** The shape to use for each node in the node pool. The shape determines the number of CPUs and the amount of memory allocated to each node. The list shows only those shapes available in your tenancy that are supported by Container Engine for Kubernetes.
      - **Quantity per Subnet:** The number of worker nodes to create for the node pool in each private subnet.
      - **Public SSH Key:** (Optional) The public key portion of the key pair you want to use for SSH access to each node in the node pool. The public key is installed on all worker nodes in the cluster. Note that if you don't specify a public SSH key, Container Engine for Kubernetes will provide one. However, since you won't have the corresponding private key, you will not have SSH access to the worker nodes. Note that because worker nodes in a 'quick cluster' are in private subnets, you cannot use SSH to access them directly (see Connecting to Worker Nodes in Private Subnets Using SSH).
      - **Kubernetes Labels:** One or more labels (in addition to a default label) to add to worker nodes in the node pool to enable the targeting of workloads at specific node pools.
   b. Either accept the defaults for the remaining cluster details, or specify alternatives in the **Additional Add Ons** panel as follows:

- **Kubernetes Dashboard Enabled:** Select if you want to use the Kubernetes Dashboard to deploy and troubleshoot containerized applications, and to manage Kubernetes resources. See Starting the Kubernetes Dashboard.
- **Tiller (Helm) Enabled:** Select if you want Tiller (the server portion of Helm) to run in the Kubernetes cluster. With Tiller running in the cluster, you can use Helm to manage Kubernetes resources.

c. (Optional) Select **View Detail Page After This Cluster Is Requested** to return to the **Cluster Details** tab (rather than the **Cluster List** page) in the Console at the end of the cluster creation process.

7. Click **Create** to create the new network resources and the new cluster.
   Container Engine for Kubernetes starts creating:
   - the network resources (such as the VCN, internet gateway, NAT gateway, route tables, security lists, private subnets), named oke-<resource-type>-quick-<cluster-name>-<creation-date>
   - the cluster, with the name you specified
   - the node pool, named pool1
   - worker nodes, with names in the format `oke-c<part-of-cluster-OCID>-n<part-of-node-pool-OCID>-s<part-of-subnet-OCID>-<slot>`

   Note that if the cluster is not created successfully for some reason (for example, if you have insufficient permissions or if you've exceeded the cluster limit for the tenancy), any network resources created during the cluster creation process are not deleted automatically. You will have to manually delete any such unused network resources.
8. Click **Close** to return to the Console.

   If you selected **View Detail Page After This Cluster Is Requested**, you return to the **Cluster Details** tab in the console. If you didn't select **View Detail Page After This Cluster Is Requested**, you return to the **Cluster List** page.

   Initially, the new cluster appears in the Console with a status of Creating. When the cluster has been created, it has a status of Active.

Container Engine for Kubernetes also creates a Kubernetes kubeconfig configuration file that you use to access the cluster using kubectl and the Kubernetes Dashboard.

Using the Console to create a 'Custom Cluster' with Explicitly Defined Settings

To create a 'custom cluster' with explicitly defined settings and existing network resources using Container Engine for Kubernetes:

1. In the Console, open the navigation menu. Under **Solutions, Platform and Edge**, go to **Developer Services** and click **Container Clusters**.
2. Choose a **Compartment** you have permission to work in, and in which you want to create the new cluster.
3. On the **Cluster List** page, click **Create Cluster**.
4. Specify configuration details for the new cluster:
    - **Name:** The name of the new cluster. Either accept the default name or enter a name of your choice. Avoid entering confidential information.
    - **Kubernetes Version:** The version of Kubernetes to run on the master nodes of the cluster. Either accept the default version or select a version of your choice. Amongst other things, the Kubernetes version you select determines the default set of admission controllers that are turned on in the created cluster (the set follows the recommendation given in the Kubernetes documentation for that version).
5. Select **Custom Create** to create a new cluster by explicitly defining the new cluster's properties and which existing network resources to use.
6. Specify the existing network resources to use for the new cluster in the **Network Selection** panel:
    - **Network Compartment:** The compartment in which the existing network resources reside.
    - **VCN:** The existing virtual cloud network that has been configured for cluster creation and deployment. See VCN Configuration.

- **Kubernetes Service LB Subnets:** Optionally, the existing subnets that have been configured to host load balancers. Load balancer subnets must be different from worker node subnets, and can be public or private. You don't have to specify any load balancer subnets. However, if you do specify load balancer subnets, the number of load balancer subnets to specify depends on the region in which you are creating the cluster:
    - If you are creating a cluster in a region with three availability domains, you can specify zero or two load balancer subnets. If you specify two load balancer subnets, the two load balancer subnets must be in different availability domains.
    - If you are creating a cluster in a region with a single availability domain, you can specify zero or one load balancer subnet.

  Note that Container Engine for Kubernetes does not yet support regional subnets, so you cannot currently select a regional subnet when specifying a load balancer subnet.

  See Subnet Configuration.
- **Kubernetes Service CIDR Block:** The available group of network addresses that can be exposed as Kubernetes services (ClusterIPs), expressed as a single, contiguous IPv4 CIDR block. For example, 10.96.0.0/16. The CIDR block you specify must not overlap with the CIDR block for the VCN. See CIDR Blocks and Container Engine for Kubernetes.
- **Pods CIDR Block:** The available group of network addresses that can be allocated to pods running in the cluster, expressed as a single, contiguous IPv4 CIDR block. For example, 10.244.0.0/16. The CIDR block you specify must not overlap with the CIDR blocks for subnets in the VCN, and can be outside the VCN CIDR block. See CIDR Blocks and Container Engine for Kubernetes.

7. Specify remaining details for the cluster in the **Additional Add Ons** panel:
    - **Kubernetes Dashboard Enabled:** Select if you want to use the Kubernetes Dashboard to deploy and troubleshoot containerized applications, and to manage Kubernetes resources. See Starting the Kubernetes Dashboard.

- **Tiller (Helm) Enabled:** Select if you want Tiller (the server portion of Helm) to run in the Kubernetes cluster. With Tiller running in the cluster, you can use Helm to manage Kubernetes resources.

8. Click **Continue**.
9. (Optional) Specify configuration details for the first node pool in the cluster in the **Node Pool** panel:
    - **Name:** A name of your choice for the new node pool. Avoid entering confidential information.
    - **Version:** The version of Kubernetes to run on each worker node in the node pool. By default, the version of Kubernetes specified for the master nodes is selected. The Kubernetes version on worker nodes must be either the same version as that on the master nodes, or an earlier version that is still compatible. See Kubernetes Versions and Container Engine for Kubernetes.
    - **Image:** The image to use on each node in the node pool. An image is a template of a virtual hard drive that determines the operating system and other software for the node.
    - **Shape:** The shape to use for each node in the node pool. The shape determines the number of CPUs and the amount of memory allocated to each node. The list shows only those shapes available in your tenancy that are supported by Container Engine for Kubernetes.
    - **Subnets:** One or more subnets configured to host worker nodes. If you specified load balancer subnets, the worker node subnets must be different. The subnets you specify can be public or private. Note that Container Engine for Kubernetes does not yet support regional subnets, so you cannot currently select a regional subnet when specifying a worker node subnet. See Subnet Configuration.
    - **Quantity per Subnet:** The number of worker nodes to create for the node pool in each subnet.
    - **Public SSH Key:** (Optional) The public key portion of the key pair you want to use for SSH access to each node in the node pool. The public key is installed on all worker nodes in the cluster. Note that if you don't specify a public SSH key,

Container Engine for Kubernetes will provide one. However, since you won't have the corresponding private key, you will not have SSH access to the worker nodes. Note that you cannot use SSH to access directly any worker nodes in private subnets (see Connecting to Worker Nodes in Private Subnets Using SSH).

- **Kubernetes Labels:** One or more labels (in addition to a default label) to add to worker nodes in the node pool to enable the targeting of workloads at specific node pools.

10. (Optional) Click **Add node pool** and specify configuration details for a second and subsequent node pools in the cluster.
    If you define multiple node pools in a cluster, you can host all of them on a single subnet. However, it's recommended best practice to host different node pools for a cluster on different subnets, one in each availability domain in the region.
11. Click **Review** to confirm the resources that will be used and created.
12. (Optional) Select **View Detail Page After This Cluster Is Requested** to return to the **Cluster Details** tab (rather than the **Cluster List** page) in the Console at the end of the cluster creation process.
13. Click **Create** to create the new cluster.
    Container Engine for Kubernetes starts creating the cluster with the name you specified.
    If you specified details for one or more node pools, Container Engine for Kubernetes creates:
    - node pools with the names you specified
    - worker nodes with names in the format `oke-c<part-of-cluster-OCID>-n<part-of-node-pool-OCID>-s<part-of-subnet-OCID>-<slot>`

    If you selected **View Detail Page After This Cluster Is Requested**, you return to the **Cluster Details** tab in the console. If you didn't select **View Detail Page After This Cluster Is Requested**, you return to the **Cluster List** page.
14. Initially, the new cluster appears in the Console with a status of Creating. When the cluster has been created, it has a status of Active.
    Container Engine for Kubernetes also creates a Kubernetes kubeconfig configuration file that you use to access the cluster using kubectl and the Kubernetes Dashboard.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the CreateCluster operation to create a cluster.

Downloading a kubeconfig File to Enable Cluster Access

When you create a cluster, you need to download a Kubernetes configuration file (commonly known as a 'kubeconfig' file) for the cluster. The kubeconfig file (by default named `config` and stored in the `$HOME/.kube` directory) provides the necessary details to access the cluster using kubectl and the Kubernetes Dashboard.

You have to follow a number of steps to download the kubeconfig file. Having completed the steps, you can start using kubectl and the Kubernetes Dashboard to manage the cluster.

To download the kubeconfig file:

Step 1: Generate an API signing key pair

If you already have an API signing key pair, go straight to the next step. If not:

1. Use OpenSSL commands to generate the key pair in the required PEM format. If you're using Windows, you'll need to install Git Bash for Windows and run the commands with that tool. See How to Generate an API Signing Key.
2. Copy the contents of the public key to the clipboard (you'll need to paste the value into the Console later).

Step 2: Upload the public key of the API signing key pair

1. In the top-right corner of the Console, open the **User** menu () and then click **User Settings** to view the details.

2. Click **Add Public Key**.
3. Paste the public key's value into the window and click **Add**.
   The key is uploaded and its fingerprint is displayed (for example, d1:b2:32:53:d3:5f:cf:68:2d:6f:8b:5f:77:8f:07:13).

Step 3: Install and configure the Oracle Cloud Infrastructure CLI

1. Install the Oracle Cloud Infrastructure CLI. See Quickstart.
2. Configure the Oracle Cloud Infrastructure CLI. See Configuration.

Step 4: Download the kubeconfig file

1. In the Console, open the navigation menu. Under **Solutions, Platform and Edge**, go to **Developer Services** and click **Container Clusters**.
2. Choose a **Compartment** you have permission to work in.
3. On the **Cluster List** page, click the name of the cluster you want to access using kubectl and the Kubernetes Dashboard. The **Cluster** page shows details of the cluster.
4. Click the **Access Kubeconfig** button to display the **How to Access Kubeconfig** dialog box.
5. Create a directory to contain the kubeconfig file. By default, the expected directory name is `$HOME/.kube`.
   For example, on Linux, enter the following command (or copy and paste it from the **How to Access Kubeconfig** dialog box):

   ```
   $ mkdir -p $HOME/.kube
   ```

6. Run the Oracle Cloud Infrastructure CLI command to download the kubeconfig file and save it in a location accessible to kubectl and the Kubernetes Dashboard.

For example, on Linux, enter the following command (or copy and paste it from the **How to Access Kubeconfig** dialog box):

```
$ oci ce cluster create-kubeconfig --cluster-id ocid1.cluster.oc1.phx.aaaaaaaaae... --file
$HOME/.kube/config  --region us-phoenix-1
```

where ocid1.cluster.oc1.phx.aaaaaaaaae... is the OCID of the current cluster. For convenience, the command in the **How to Access Kubeconfig** dialog box already includes the cluster's OCID.

Note that if a kubeconfig file already exists in the location you specify, details about the cluster will be added as a new context to the existing kubeconfig file. The `current-context:` element in the kubeconfig file will be set to point to the newly added context.

7. Set the value of the KUBECONFIG environment variable to point to the name and location of the kubeconfig file. For example, on Linux, enter the following command (or copy and paste it from the **How to Access Kubeconfig** dialog box):

```
$ export KUBECONFIG=$HOME/.kube/config
```

Step 5: Verify that kubectl is available

1. Verify that kubectl is available by entering the following command:

```
$ kubectl version
```

Information about the versions of Kubernetes running in the cluster is shown.

2. Verify that kubectl can connect to the cluster by entering the following command:

```
$ kubectl get nodes
```

Information about the nodes in the cluster is shown.

You can now use kubectl and the Kubernetes Dashboard to perform operations on the cluster.



Note

The kubeconfig file contains credentials to access the cluster, including a hashed version of the private key component of your API signing key pair. The kubeconfig file is therefore specific to you, and is valid for as long as your API signing key pair is valid. Best practice is not to share the kubeconfig file with other users. If you do share the kubeconfig file with other users, be aware that those users will be able to access all the clusters to which you have access in the tenancy.

Modifying a Kubernetes Cluster

You can use Container Engine for Kubernetes to modify the node pool details of existing Kubernetes clusters.

You can change:

- the name of the node pool
- the number of node pools in a cluster by adding new node pools, or deleting existing node pools
- the number of worker nodes in a node pool by changing:
    - the number of subnets used by a node pool
    - the number of worker nodes in each subnet
- the version of Kubernetes to run on new worker nodes

However, note that you cannot change:

- the name of the cluster
- the shape of existing worker nodes

- the operating system running on existing worker nodes
- the version of Kubernetes running on existing worker nodes

Using the Console

To modify an existing Kubernetes cluster:

1. In the Console, open the navigation menu. Under **Solutions, Platform and Edge**, go to **Developer Services** and click **Container Clusters**.
2. Choose a **Compartment** you have permission to work in.
3. On the **Cluster List** page, click the name of the cluster you want to modify.
4. Use the buttons across the top of the **Cluster** page as follows:
    - If you want to download the kubeconfig configuration file for the cluster, click the **Access Kubeconfig** button (see Downloading a kubeconfig File to Enable Cluster Access).
    - If you want to add a new node pool to the cluster, click the **Add Node Pool** button and enter details for the new node pool.
    - If you want to delete the cluster along with its master nodes and worker nodes, click the **Delete Cluster** button.
    - If a newer version of Kubernetes is available than the one running on the master nodes in the cluster, the **Upgrade Available** button is enabled. If you want to upgrade the master nodes to a newer version, click **Upgrade Available** (see Upgrading the Version of Kubernetes Running on Master Nodes).
5. Use the **Cluster Details** tab to see information about the cluster, including:
    - The status of the cluster, and of the node pools in the cluster.
    - The cluster's OCID.
    - The Kubernetes version running on the master nodes in the cluster.
    - The address of the Kubernetes endpoint.

6. Use the **Node Pools** tab to:
    - View a summary of each node pool and the worker nodes within it.
    - Change the name of a node pool by selecting **Edit** from the **Actions** menu.
    - Change the number of worker nodes in a node pool by selecting **Scale** from the **Actions** menu and specifying the number of subnets used by a node pool and the number of worker nodes in each subnet.
    - View and edit configuration details of specific worker nodes by selecting **Show Node Details** and clicking the name of the worker node.
    - Delete a node pool by selecting **Delete Node Pool** from the **Actions** menu.
7. Use the **Getting Started** tab to:
    - View and copy the commands to start the Kubernetes Dashboard to view the deployed application running on nodes in the cluster (see Starting the Kubernetes Dashboard).
    - View and copy the commands to download and deploy a sample nginx application using the Kubernetes command line tool kubectl from the instructions in a manifest file (see Deploying a Sample Nginx App on a Cluster Using kubectl).

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the Update Cluster and the UpdateNodePool operations to modify an existing Kubernetes cluster.

Deleting a Kubernetes Cluster

You can delete a cluster along with its master nodes and worker nodes.

Using the Console

To delete a Kubernetes cluster using Container Engine for Kubernetes:

1. In the Console, open the navigation menu. Under **Solutions, Platform and Edge**, go to **Developer Services** and click **Container Clusters**.
2. Choose a **Compartment** you have permission to work in.
3. On the **Cluster List** page, click the Delete icon beside the name of the cluster to delete, and confirm that you want to delete it.

You can also delete a cluster using the **Delete Cluster** button on the **Cluster** page.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the DeleteCluster operation to delete a cluster.

Monitoring Clusters

Having created a cluster, you can monitor the status of the cluster itself, and the nodes and node pools within it.

Using the Console

To monitor a Kubernetes cluster:

1. In the Console, open the navigation menu. Under **Solutions, Platform and Edge**, go to **Developer Services** and click **Container Clusters**.
2. Choose a **Compartment** you have permission to work in.

The **Status** column on the **Cluster List** page shows a summary status for each individual cluster and its master nodes. Clusters can have one of the following statuses:

Cluster Status	Explanation	Possible Reason
Creating	Cluster is in the process of being created.	Application is being deployed.
Active	Cluster is running normally.	Master nodes are running normally.
Failed	Cluster is not running due to an unrecoverable error.	Possible reasons: • a problem setting up load balancers • an error installing cluster add-ons (Tiller, Kubernetes dashboard) • conflicts in networking ranges
Deleting	Cluster is in the process of being deleted. Application no longer required, so resources in the process of being released.	Application no longer required, so resources in the process of being released.
Deleted	Cluster has been deleted. Application no longer required, so resources have been released.	Application no longer required, so resources have been released.
Updating	Version of Kubernetes on the master nodes is in the process of being upgraded.	A newly supported version of Kubernetes has become available.

Note that the cluster's summary status is not necessarily directly related to the status of node pools and nodes within the cluster.

3. On the **Cluster List** page, click the name of the cluster for which you want to see detailed status.
   The **Cluster Details** tab shows the summary status for the cluster and its master nodes.
4. Use the **Node Pools** tab to see the status of individual nodes within each node pool.
   Nodes can have one of the following statuses:

Node Status	Explanation	Possible Reason
Creating	Node is being created.	Compute instance in the process of being created.
Active	Node is running normally.	Node is running normally.
Updating	Node is in the process of being updated.	Container Engine for Kubernetes is performing an operation on the node.
Deleting	Node is in the process of being deleted.	Application no longer required, so resources in the process of being released.
Deleted	Node has been deleted.	Application no longer required, so resources have been released.
Inactive	Node still exists, but is not running.	Compute resource has a status of Stopped, Stopping, or Down For Maintenance.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line

Interface.

Use the GetCluster and GetNodePool operations to monitor the status of Kubernetes clusters.

Monitoring Operations of Container Engine for Kubernetes

You can monitor operations performed by Container Engine for Kubernetes as follows:

- You can monitor and manage operations performed on a particular cluster by Container Engine for Kubernetes using the Work Requests tab of the cluster's Summary page.
- You can view all operations performed by Container Engine for Kubernetes as log events using the Oracle Cloud Infrastructure Audit service.

Using the Console

To monitor and manage operations performed on a particular cluster:

1. In the Console, open the navigation menu. Under **Solutions, Platform and Edge**, go to **Developer Services** and click **Container Clusters**.
2. Choose a **Compartment** you have permission to work in.
3. On the **Cluster List** page, click the name of the cluster for which you want to monitor and manage operations.
   The **Cluster** page shows information about the cluster.

To view all operations performed by Container Engine for Kubernetes as log events:

1. In the Console, open the navigation menu. Under **Governance and Administration**, go to **Governance** and click **Policies**.
2. Choose a **Compartment** you have permission to work in.
3. Search and filter to show the operations performed by Container Engine for Kubernetes. See Viewing Audit Log Events.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the GetWorkRequest, DeleteWorkRequest, ListWorkRequestErrors, ListWorkRequestLogs, and ListWorkRequests operations to monitor and manage operations performed by Container Engine for Kubernetes.

Accessing a Cluster Using kubectl

You can use the Kubernetes command line tool kubectl to perform operations on a cluster you've created with Container Engine for Kubernetes. Before you can use kubectl you need to specify the cluster on which to perform operations.

To access a cluster using kubectl:

1. If you haven't already done so, install kubectl (see the kubectl documentation).
2. If you haven't already done so, download the cluster's kubeconfig configuration file and set the KUBECONFIG environment variable to point to the file. See Downloading a kubeconfig File to Enable Cluster Access.
3. In a terminal window, enter `kubectl` followed by the command for the operation you want to perform on the cluster. For a list of available commands and options, see the kubectl documentation.
   Note that you must have the appropriate permissions to run the command you enter. See About Access Control and Container Engine for Kubernetes.

Starting the Kubernetes Dashboard

Kubernetes Dashboard is a web-based user interface that you can use as an alternative to the Kubernetes kubectl command line tool to:

- deploy containerized applications to a Kubernetes cluster
- troubleshoot your containerized applications

You use the Kubernetes Dashboard to get an overview of applications running on a cluster, as well as to create or modify individual Kubernetes resources. The Kubernetes Dashboard also reports the status of Kubernetes resources in the cluster, and any errors that have occurred. Note that to use the Kubernetes Dashboard, it must have been enabled when the cluster was initially created.

In contrast to the Kubernetes Dashboard, Container Engine for Kubernetes enables you to create and delete Kubernetes clusters and node pools, and to manage the associated compute, network, and storage resources.

Before you can use the Kubernetes Dashboard, you need to specify the cluster on which to perform operations.

To start the Kubernetes Dashboard:

1. If you haven't already done so, download the cluster's kubeconfig configuration file and set the KUBECONFIG environment variable to point to the file. See Downloading a kubeconfig File to Enable Cluster Access.
2. In a terminal window, enter `kubectl proxy` to start the Kubernetes Dashboard.
3. Open a browser and go to `http://localhost:8001/api/v1/namespaces/kube-system/services/https:kubernetes-dashboard:/proxy/` to display the Kubernetes Dashboard.
4. Click **Overview** to see the applications deployed on the cluster.

Deploying a Sample Nginx App on a Cluster Using kubectl

Having created a Kubernetes cluster using Container Engine for Kubernetes, you'll typically want to try it out by deploying an application on the nodes in the cluster. For convenience, the **Cluster** page includes a **Getting Started** tab that makes it easy to view and copy the commands to:

- download the kubeconfig configuration file for the cluster
- download and deploy a sample Nginx application using the Kubernetes command line tool kubectl from the instructions in a manifest file

- start the Kubernetes Dashboard to view the deployed application running on nodes in the cluster

To deploy the sample nginx application:

1. If you haven't already done so, download the cluster's kubeconfig configuration file and set the KUBECONFIG environment variable to point to the file. See Downloading a kubeconfig File to Enable Cluster Access.
2. In a terminal window, deploy the sample Nginx application by entering `kubectl create -f https://k8s.io/docs/tasks/run-application/deployment.yaml`



Tip

If the command fails to connect to `https://k8s.io/docs/tasks/run-application/deployment.yaml` , go to the url in a browser and download the manifest file `deployment.yaml` to a local directory. Repeat the `kubectl create` command and specify the local location of the `deployment.yaml` file.

3. Use the Kubernetes Dashboard or kubectl to confirm that the sample application has deployed successfully. For example:
    a. Enter `kubectl proxy` to start the Kubernetes Dashboard.
    b. Open a browser and go to `http://localhost:8001/api/v1/namespaces/kube-system/services/https:kubernetes-dashboard:/proxy/` to display the Kubernetes Dashboard.
    c. Click **Overview** to see the applications deployed on the cluster.

You can see the Nginx sample application has been deployed as two pods, on two nodes in the cluster.

Pulling Images from Registry during Deployment

During the deployment of an application to a Kubernetes cluster, you'll typically want one or more images to be pulled from a Docker registry. In the application's manifest file you specify the images to pull, the registry to pull them from, and the credentials to use when pulling the images. The manifest file is commonly also referred to as a pod spec, or as a deployment.yaml file (although other filenames are allowed).

If you want the application to pull images that reside in Oracle Cloud Infrastructure Registry, you have to perform two steps:

- You have to use kubectl to create a Docker registry secret. The secret contains the Oracle Cloud Infrastructure credentials to use when pulling the image. When creating secrets, Oracle strongly recommends you use the latest version of kubectl (see the kubectl documentation).
- You have to specify the image to pull from Oracle Cloud Infrastructure Registry, including the repository location and the Docker registry secret to use, in the application's manifest file.

To create a Docker registry secret:

1. If you haven't already done so, download the cluster's kubeconfig configuration file and set the KUBECONFIG environment variable to point to the file. See Downloading a kubeconfig File to Enable Cluster Access.
2. In a terminal window, enter:

   ```
   $ kubectl create secret docker-registry <secret-name> --docker-server=<region-code>.ocir.io --docker-username='<tenancy-name>/<oci-username>' --docker-password='<oci-auth-token>' --docker-email='<email-address>'
   ```

   where:

   - `<secret-name>` is a name of your choice, that you will use in the manifest file to refer to the secret . For example, `ocirsecret`
   - `<region-code>` is the code for the Oracle Cloud Infrastructure Registry region you're using. For example, `iad`. See Availability by Region Name and Region Code for the list of region codes.

- `ocir.io` is the Oracle Cloud Infrastructure Registry name.
- `<tenancy-name>` is the tenancy containing the repository from which the application is to pull the image. For example, `acme-dev`
- `<oci-username>` is the username to use when pulling the image. The username must have access to the tenancy specified by `<tenancy-name>`. For example, `jdoe@acme.com` . If your tenancy is federated with Oracle Identity Cloud Service, use the format `oracleidentitycloudservice/<username>`
- `<oci-auth-token>` is the auth token of the user specified by `<oci-username>`. For example, `k]j64r{1sJSSF-;)K8`
- `<email-address>` is an email address. An email address is required, but it doesn't matter what you specify. For example, `jdoe@acme.com`

Note the use of single quotes around strings containing special characters.

For example, combining the previous examples, you might enter:

```
$ kubectl create secret docker-registry ocirsecret --docker-server=phx.ocir.io --docker-
username='acme-dev/jdoe@acme.com' --docker-password='k]j64r{1sJSSF-;)K8' --docker-
email='jdoe@acme.com'
```

Having created the Docker secret, you can now refer to it in the application manifest file.

To specify the image to pull from Oracle Cloud Infrastructure Registry, along with the Docker secret to use, during deployment of an application to a cluster:

1. Open the application's manifest file in a text editor.
2. Add the following sections to the manifest file:
    a. Add a `containers` section that specifies the name and location of the container you want to pull from Oracle Cloud Infrastructure Registry, along with other deployment details.
    b. Add an `imagePullSecrets` section to the manifest file that specifies the name of the Docker secret you created to access the Oracle Cloud Infrastructure Registry.

Here's an example of what the manifest might look like when you've added the `containers` and `imagePullSecrets` sections:

```
apiVersion: v1
kind: Pod
metadata:
  name: ngnix-image
spec:
  containers:
    - name: ngnix
      image: phx.ocir.io/acme-dev/project01/ngnix-lb:latest
      imagePullPolicy: Always
      ports:
      - name: nginx
        containerPort: 8080
        protocol: TCP
  imagePullSecrets:
    - name: ocirsecret
```

3. Save and close the manifest file.

Connecting to Worker Nodes Using SSH

If you provided a public SSH key when creating the node pool in a cluster, the public key is installed on all worker nodes in the cluster. On UNIX and UNIX-like platforms (including Solaris and Linux), you can then connect through SSH to the worker nodes using the ssh utility (an SSH client) to perform administrative tasks.

Note the following instructions assume the UNIX machine you use to connect to the worker node:

- Has the ssh utility installed.
- Has access to the SSH private key file paired with the SSH public key that was specified when the cluster was created.

How to connect to worker nodes using SSH depends on whether you specified public or private subnets for the worker nodes when defining the node pools in the cluster.

Connecting to Worker Nodes in Public Subnets Using SSH

Before you can connect to a worker node in a public subnet using SSH, you must define an ingress rule in the subnet's security list to allow SSH access. The ingress rule must allow access to port 22 on worker nodes from source 0.0.0.0/0 and any source port, as follows:

Type	Source CIDR	IP Protocol	Source Port Range	Dest. Port Range	Type and Code	Allows: and Description:
Stateful	0.0.0.0/0	TCP	All	22	n/a	**Allows:** TCP traffic for ports: 22 SSH Remote Login Protocol **Description:** Enables SSH access.

To connect to a worker node in a public subnet through SSH from a UNIX machine using the ssh utility:

1. Find out the IP address of the worker node to which you want to connect. You can do this in a number of ways:
    - Using kubectl. If you haven't already done so, download the cluster's kubeconfig configuration file and set the KUBECONFIG environment variable to point to the file. See Downloading a kubeconfig File to Enable Cluster Access. Then in a terminal window, enter `kubectl get nodes` to see the public IP addresses of worker nodes in node pools in the cluster.
    - Using the Console. In the Console, display the **Cluster List** page and then select the cluster to which the worker node belongs. Click **Node Pools** to see the public IP addresses of worker nodes in every node pool in the cluster.
    - Using the REST API. Use the ListNodePools operation to see the public IP addresses of worker nodes in a node pool.
2. In the terminal window, enter `ssh opc@<node_ip_address>` to connect to the worker node, where `<node_ip_address>` is the IP address of the worker node that you made a

note of earlier. For example, you might enter `ssh opc@192.0.2.254`.

Note that if the SSH private key is not stored in the file or in the path that the ssh utility expects (for example, the ssh utility might expect the private key to be stored in ~/.ssh/id_rsa), you must explicitly specify the private key filename and location in one of two ways:

- Use the `-i` option to specify the filename and location of the private key. For example, `ssh -i ~/.ssh/my_keys/my_host_key_filename opc@192.0.2.254`
- Add the private key filename and location to an SSH configuration file, either the client configuration file (~/.ssh/config) if it exists, or the system-wide client configuration file (/etc/ssh/ssh_config). For example, you might add the following:

```
Host 192.0.2.254 IdentityFile ~/.ssh/my_keys/my_host_key_filename
```

For more about the ssh utility's configuration file, enter `man ssh_config`

Note also that permissions on the private key file must allow you read/write/execute access, but prevent other users from accessing the file. For example, to set appropriate permissions, you might enter `chmod 600 ~/.ssh/my_keys/my_host_key_filename`. If permissions are not set correctly and the private key file is accessible to other users, the ssh utility will simply ignore the private key file.

Connecting to Worker Nodes in Private Subnets Using SSH

Worker nodes in private subnets have private IP addresses only (they do not have public IP addresses). They can only be accessed by other resources inside the VCN. Oracle recommends using bastion hosts to control external access (such as SSH) to worker nodes in private subnets. A bastion host is in a public subnet, has a public IP address, and is accessible from the internet. For more information about bastion hosts, see the white paper Bastion Hosts: Protected Access for Virtual Cloud Networks.

About Access Control and Container Engine for Kubernetes

To perform operations on a Kubernetes cluster, you must have appropriate permissions to access the cluster.

For most operations on Kubernetes clusters created and managed by Container Engine for Kubernetes, Oracle Cloud Infrastructure Identity and Access Management (IAM) provides access control. A user's permissions to access clusters comes from the groups to which they belong. The permissions for a group are defined by policies. Policies define what actions members of a group can perform, and in which compartments. Users can then access clusters and perform operations based on the policies set for the groups they are members of.

IAM provides control over:

- whether a user can create or delete clusters
- whether a user can add, remove, or modify node pools
- which Kubernetes object create/delete/view operations a user can perform on all clusters within a compartment or tenancy

See Policy Configuration for Cluster Creation and Deployment.

In addition to IAM, the Kubernetes RBAC Authorizer can enforce additional fine-grained access control for users on specific clusters via Kubernetes RBAC roles and clusterroles. A Kubernetes RBAC role is a collection of permissions. For example, a role might include read permission on pods and list permission for pods. A Kubernetes RBAC clusterrole is just like a role, but can be used anywhere in the cluster. A Kubernetes RBAC rolebinding maps a role to a user or set of users, granting that role's permissions to those users for resources in that namespace. Similarly, a Kubernetes RBAC clusterrolebinding maps a clusterrole to a user or set of users, granting that clusterrole's permissions to those users across the entire cluster.

IAM and the Kubernetes RBAC Authorizer work together to enable users who have been successfully authorized by at least one of them to complete the requested Kubernetes operation.

When a user attempts to perform any operation on a cluster (except for create role and create clusterrole operations), IAM first determines whether the group to which the user belongs has

the appropriate and sufficient permissions. If so, the operation succeeds. If the attempted operation also requires additional permissions granted via a Kubernetes RBAC role or clusterrole, the Kubernetes RBAC Authorizer then determines whether the user has been granted the appropriate Kubernetes role or clusterrole.

Typically, you'll want to define your own Kubernetes RBAC roles and clusterroles when deploying a Kubernetes cluster to provide additional fine-grained control. When you attempt to perform a create role or create clusterrole operation, the Kubernetes RBAC Authorizer first determines whether you have sufficient Kubernetes privileges. To create a role or clusterrole, you must have been assigned an existing Kubernetes RBAC role (or clusterrole) that has at least the same or higher privileges as the new role (or clusterrole) you're attempting to create.

By default, users are not assigned any Kubernetes RBAC roles (or clusterroles). So before attempting to create a new role (or clusterrole), you must be assigned an appropriately privileged role (or clusterrole). A number of such roles and clusterroles are always created by default, including the cluster-admin clusterrole (for a full list, see Default Roles and Role Bindings in the Kubernetes documentation). The cluster-admin clusterrole essentially confers super-user privileges. A user granted the cluster-admin clusterrole can perform any operation across all namespaces in a given cluster.

Example: Granting the Kubernetes RBAC cluster-admin clusterrole



Note

The following instructions assume:

- You're in the tenancy's Administrators group, and therefore have the required permissions to access clusters.
- You have the Kubernetes RBAC cluster-admin clusterrole, and therefore have the required



access to create Kubernetes RBAC roles and clusterroles.

Follow these steps to grant a user the Kubernetes RBAC cluster-admin clusterrole on a cluster deployed on Oracle Cloud Infrastructure:

1. If you haven't already done so, download the cluster's kubeconfig configuration file and set the KUBECONFIG environment variable to point to the file. See Downloading a kubeconfig File to Enable Cluster Access.
2. In a terminal window, grant the Kubernetes RBAC cluster-admin clusterrole to the user by entering:

   ```
   $ kubectl create clusterrolebinding <my-cluster-admin-binding> --clusterrole=cluster-admin --
   user=<user_OCID>
   ```

   where:

   - `<my-cluster-admin-binding>` is a string of your choice to be used as the name for the binding between the user and the Kubernetes RBAC cluster-admin clusterrole. For example, `jdoe_clst_adm`
   - `<user_OCID>` is the user's OCID (obtained from the Console). For example, `ocid1.user.oc1..aaaaa...zutq` (abbreviated for readability).

   For example:

   ```
   $ kubectl create clusterrolebinding jdoe_clst_adm --clusterrole=cluster-admin --
   user=ocid1.user.oc1..aaaaa...zutq
   ```

Example: Giving a developer user the ability to read pods in a new cluster



Note

The following instructions assume:



- You're in the tenancy's Administrators group, and therefore have the required permissions to create clusters, and to manage users and groups.
- You have the Kubernetes RBAC cluster-admin clusterrole, and therefore have the required access to create Kubernetes RBAC roles and clusterroles.

Follow these steps to give a developer the necessary Oracle Cloud Infrastructure and Kubernetes RBAC permissions to use kubectl to view pods running on a cluster deployed on Oracle Cloud Infrastructure:

1. Create a new Oracle Cloud Infrastructure user for the developer to use (for example, called jdoe@acme.com), and make a note of the new user's OCID (for example, `ocid1.user.oc1..aaaaa...tx5a`, abbreviated for readability). See To create a user.
2. Create a new Oracle Cloud Infrastructure group and add the new user to the group (for example, called acme-dev-pod-vwr). See To create a group.
3. Create a new Oracle Cloud Infrastructure policy that grants the new group the CLUSTER_USE permission on clusters in the tenancy, with a policy statement like:

   ```
   Allow group acme-dev-pod-vwr to use clusters in tenancy
   ```

   See To create a policy.
4. Create a new cluster in the Console. See Creating a Kubernetes Cluster .
5. Download the cluster's kubeconfig configuration file and set the KUBECONFIG environment variable to point to the file. See Downloading a kubeconfig File to Enable Cluster Access.
6. In a text editor, create a file (for example, called role-pod-reader.yaml) with the following content. This file defines a Kubernetes RBAC role that enables users to read pod details.

   ```
   kind: Role
   apiVersion: rbac.authorization.k8s.io/v1
   ```

```
metadata:
 namespace: default
 name: pod-reader
rules:
- apiGroups: [""] # "" indicates the core API group
  resources: ["pods"]
  verbs: ["get", "watch", "list"]
```

7. In a terminal window, create the new role in the cluster using kubectl. For example, if you gave the yaml file that defines the new role the name role-pod-reader.yaml, enter the following:

```
$ kubectl create -f role-pod-reader.yaml
```

8. In a terminal window, bind the Kubernetes RBAC role you just created to the Oracle Cloud Infrastructure user account you created earlier by entering the following to create a new rolebinding (in this case, called pod-reader-binding):

```
$ kubectl create rolebinding pod-reader-binding --role=pod-reader --
user=ocid1.user.oc1..aaaaa...tx5a
```

9. Give the developer the credentials of the new Oracle Cloud Infrastructure user you created earlier, and tell the developer they can now see details of pods running on the cluster deployed on Oracle Cloud Infrastructure by:
    - Signing in to the Console using the new user's credentials.
    - Following the instructions in Downloading a kubeconfig File to Enable Cluster Access to obtain their own copy of the cluster's kubeconfig file. If the file does not have the expected default name and location of `$HOME/.kube/config`, the developer will also have to set the KUBECONFIG environment variable to point to the file. The developer's copy of the kubeconfig file will contain a hashed version of the public key component of their API signing key pair.
    - Using kubectl to see details of the pods by entering:

      ```
      $ kubectl get pods
      ```

Kubernetes Versions and Container Engine for Kubernetes

When you create a new Kubernetes cluster using Container Engine for Kubernetes, you specify:

- The version of Kubernetes to run on the master nodes in the cluster.
- The version of Kubernetes to run on the worker nodes in each node pool. All worker nodes in the same node pool run the same version of Kubernetes. Different node pools in a cluster can run different versions of Kubernetes.

The version of Kubernetes that you specify for the worker nodes in a node pool must be either the same Kubernetes version as that running on the master nodes, or an earlier Kubernetes version that is still compatible. In other words:

- The master nodes in a new cluster must run the same version of Kubernetes as the version running on worker nodes, or must be no more than two versions ahead.
- The worker nodes in a node pool must not run a more recent version of Kubernetes than the associated master nodes.

About Kubernetes Versions

New versions of Kubernetes are released periodically that contain new features and bug fixes.

Kubernetes version numbers have the format `x.y.z` where `x` is a major release, `y` is a minor release, and `z` is a patch release. For example, 1.12.7.

Kubernetes itself is supported for three minor versions at a time (the current release version and two previous versions).

As described in the Kubernetes documentation, a certain amount of version variation is permissible between master nodes and worker nodes in a cluster:

- The Kubernetes version on worker nodes can lag behind the version on the master nodes by up to two versions, but no more. If the version on the worker nodes is more than two versions behind the version on the master nodes, the Kubernetes versions on the worker nodes and the master nodes are incompatible.

- The Kubernetes version on worker nodes must never be more recent than the version on the master nodes.

About Upgrading Clusters to Newer Kubernetes Versions

After a new version of Kubernetes has been released and when Container Engine for Kubernetes supports the new version, you can use Container Engine for Kubernetes to upgrade master nodes running older versions of Kubernetes. Because Container Engine for Kubernetes distributes the Kubernetes Control Plane on multiple Oracle-managed master nodes (distributed across different availability domains in a region where supported) to ensure high availability, you're able to upgrade the Kubernetes version running on master nodes with zero downtime.

Having upgraded master nodes to a new version of Kubernetes, you can subsequently create new node pools running the newer version. Alternatively, you can continue to create new node pools that will run older versions of Kubernetes (providing those older versions are compatible with the Kubernetes version running on the master nodes).

Note that you upgrade master nodes by performing an ‘in-place’ upgrade, but you upgrade worker nodes by performing an ‘out-of-place’ upgrade. To upgrade the version of Kubernetes running on worker nodes in a node pool, you replace the original node pool with a new node pool that has new worker nodes running the appropriate Kubernetes version. Having 'drained' existing worker nodes in the original node pool to prevent new pods starting and to delete existing pods, you can then delete the original node pool.

Also note the following:

- Container Engine for Kubernetes only upgrades the Kubernetes version running on master nodes when you explicitly initiate the upgrade operation.
- After upgrading master nodes to a newer version of Kubernetes, you cannot downgrade the master nodes to an earlier Kubernetes version.
- Before you upgrade the version of Kubernetes running on the master nodes, it is your responsibility to test that applications deployed on the cluster are compatible with the new Kubernetes version. For example, before upgrading the existing cluster, you might create a new separate cluster with the new Kubernetes version to test your applications.

- The versions of Kubernetes running on the master nodes and the worker nodes must be compatible (that is, the Kubernetes version on the master nodes must be no more than two minor versions ahead of the Kubernetes version on the worker nodes).
- If the version of Kubernetes currently running on the master nodes is more than one version behind the most recent supported version, you are given a choice of versions to upgrade to. If you want to upgrade to a version of Kubernetes that is more than one version ahead of the version currently running on the master nodes, you must upgrade to each intermediate version in sequence without skipping versions. For example, if you want to upgrade the master nodes from Kubernetes version 1.10.11 to Kubernetes version 1.12.7, you must first upgrade the master nodes from version 1.10.11 to version 1.11.9, and then from version 1.11.9 to version 1.12.7.

Kubernetes Versions Supported by Container Engine for Kubernetes

Container Engine for Kubernetes supports the following versions of Kubernetes:

Kubernetes Version	Supported by Container Engine for Kubernetes?	Notes
1.10.x and earlier	No	N/A
1.11.9	Yes	Note that versions of Kubernetes 1.11.x prior to 1.11.9 (for example, 1.11.8) are no longer supported. You cannot: • create clusters running earlier 1.11.x versions • add new node pools to existing clusters running earlier 1.11.x versions If you do have clusters running 1.11.x versions earlier than 1.11.9, Oracle strongly recommends you upgrade those clusters to version 1.11.9 or version 1.12.7.

Kubernetes Version	Supported by Container Engine for Kubernetes?	Notes
1.12.7	Yes	Note that versions of Kubernetes 1.12.x prior to 1.12.7 (for example, 1.12.6) are no longer supported. You cannot: • create clusters running earlier 1.12.x versions • add new node pools to existing clusters running earlier 1.12.x versions If you do have clusters running 1.12.x versions earlier than 1.12.7, Oracle strongly recommends you upgrade those clusters to version 1.12.7.

Upgrading the Version of Kubernetes Running on Master Nodes

When Container Engine for Kubernetes supports a newer version of Kubernetes than the version currently running on the master nodes in a cluster, you can upgrade the Kubernetes version running on the master nodes.

Important: After you've upgraded master nodes to a newer version of Kubernetes, you can't downgrade the master nodes to an earlier Kubernetes version. It's therefore important that before you upgrade the version of Kubernetes running on the master nodes, you test that applications deployed on the cluster are compatible with the new Kubernetes version.

Using the Console

To upgrade the version of Kubernetes running on the master nodes:

1. In the Console, open the navigation menu. Under **Solutions, Platform and Edge**, go to **Developer Services** and click **Container Clusters**.
2. Choose a **Compartment** you have permission to work in.
3. On the **Cluster List** page, click the name of the cluster where you want to upgrade the version of Kubernetes running on the master nodes.
   If a newer version of Kubernetes is available than the one running on the master nodes in the cluster, the **Upgrade Available** button is enabled at the top of the **Cluster** page.
4. Click **Upgrade Available** to upgrade the master nodes to a newer version.
5. In the **Upgrade Cluster Master** dialog box, select the version of Kubernetes to which to upgrade the master nodes, and click **Confirm**.

The version of Kubernetes running on the master nodes is upgraded. From now on, the new version of Kubernetes will appear as an option when you're defining new node pools for the cluster.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the UpdateCluster operation to upgrade the version of Kubernetes running on the master nodes.

'Upgrading' the version of Kubernetes running on worker nodes by creating a new node pool

To 'upgrade' the version of Kubernetes running on worker nodes in a node pool, you replace the original node pool with a new node pool that has new worker nodes running the appropriate Kubernetes version. Having 'drained' existing worker nodes in the original node

pool to prevent new pods starting and to delete existing pods, you can then delete the original node pool.

To 'upgrade' the version of Kubernetes on worker nodes by creating a new node pool:

1. In the Console, open the navigation menu. Under **Solutions, Platform and Edge**, go to **Developer Services** and click **Container Clusters**.
2. Choose a **Compartment** you have permission to work in.
3. On the **Cluster List** page, click the name of the cluster where you want to change the Kubernetes version running on worker nodes.
4. On the **Cluster** page, click **Add Node Pool** to create a new node pool and specify the required version of Kubernetes for its worker nodes.

   The version of Kubernetes you specify must be compatible with the version that is running on the master nodes.
5. If there are labels attached to worker nodes in the original node pool and those labels are used by selectors (for example, to determine the nodes on which to run pods), then use the `kubectl label nodes` command to attach the same labels to the new worker nodes in the new node pool. See Assigning Pods to Nodes in the Kubernetes documentation.
6. For each worker node in the original node pool, prevent new pods from starting and delete existing pods by entering `kubectl drain <node_name>` for each worker node.

   For more information:

   - about using kubectl, see Accessing a Cluster Using kubectl
   - about the drain command, see drain in the Kubernetes documentation

   Recommended: Leverage pod disruption budgets as appropriate for your application to ensure that there's a sufficient number of replica pods running throughout the drain operation.

   After all the worker nodes have been drained from the original node pool and pods are running on worker nodes in the new node pool, you can delete the original node pool.
7. On the **Cluster** page, display the **Node Pools** tab and select **Delete Node Pool** from the **Actions** menu.

   The original node pool and all its worker nodes are deleted.

Creating Load Balancers to Distribute Traffic Between Cluster Nodes

When you create a service, you can optionally create a load balancer to distribute service traffic among the nodes assigned to that service. The key fields in the configuration of a load balancer are the **type** of service being created and the **ports** that the load balancer will listen to.

Creating Load Balancers to Distribute HTTP Traffic

Consider the following configuration file, `nginx_lb.yaml`. It defines a deployment (`kind: Deployment`) for the `nginx` app, followed by a service definition with a type of LoadBalancer (`type: LoadBalancer`) that balances http traffic on port 80 for the `nginx` app.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: my-nginx
  labels:
    app: nginx
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx
        image: nginx:1.7.9
        ports:
        - containerPort: 80
---
apiVersion: v1
kind: Service
metadata:
  name: my-nginx-svc
```

```
  labels:
    app: nginx
spec:
  type: LoadBalancer
  ports:
  - port: 80
  selector:
    app: nginx
```

The first part of the configuration file defines an Nginx deployment, requesting that it be hosted on 3 pods running the nginx:1.7.9 image, and accept traffic to the containers on port 80.

The second part of the configuration file defines the Nginx service, which uses type LoadBalancer to balance Nginx traffic on port 80 amongst the available pods.

To create the deployment and service defined in `nginx_lb.yaml` while connected to your Kubernetes cluster, enter the command:

```
$ kubectl apply -f nginx_lb.yaml
```

This command outputs the following upon successful creation of the deployment and the load balancer:

```
deployment "my-nginx" created
service "my-nginx-svc" created
```

The load balancer may take a few minutes to go from a pending state to being fully operational. You can view the current state of your cluster by entering `kubectl get all`, where your output looks similar to the following:

```
$ kubectl get all

NAME                                READY     STATUS    RESTARTS   AGE
po/my-nginx-431080787-0m4m8         1/1       Running   0          3m
po/my-nginx-431080787-hqqcr         1/1       Running   0          3m
po/my-nginx-431080787-n8125         1/1       Running   0          3m

NAME               CLUSTER-IP      EXTERNAL-IP      PORT(S)         AGE
svc/kubernetes     203.0.113.1     <NONE>           443/TCP         3d
svc/my-nginx-svc   203.0.113.7     192.0.2.22       80:30269/TCP    3m

NAME                     DESIRED   CURRENT   UP-TO-DATE   AVAILABLE   AGE
```

```
deploy/my-nginx            3         3         3            3           3m

NAME                               DESIRED   CURRENT   READY     AGE
rs/my-nginx-431080787              3         3         3         3m
```

The output shows that the `my-nginx` deployment is running on 3 pods (the po/my-nginx entries), that the load balancer is running (svc/my-nginx-svc) and has an external IP (192.0.2.22) that clients can use to connect to the app that's deployed on the pods.

Creating Load Balancers with SSL Support to Distribute HTTPS Traffic

You can create a load balancer with SSL termination, allowing https traffic to an app to be distributed among the nodes in a cluster. This example provides a walkthrough of the configuration and creation of a load balancer with SSL support.

Consider the following configuration file, `nginx-demo-svc-ssl.yaml`, which defines an Nginx deployment and exposes it via a load balancer that serves http on port 80, and https on port 443. This sample creates an Oracle Cloud Infrastructure load balancer, by defining a service with a type of LoadBalancer (`type: LoadBalancer`).

```
apiVersion: apps/v1beta1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  replicas: 2
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx
        image: nginx
        ports:
        - containerPort: 80
---
kind: Service
apiVersion: v1
metadata:
```

```
  name: nginx-service
  annotations:
    service.beta.kubernetes.io/oci-load-balancer-ssl-ports: "443"
    service.beta.kubernetes.io/oci-load-balancer-tls-secret: ssl-certificate-secret
spec:
  selector:
    app: nginx
  type: LoadBalancer
  ports:
  - name: http
    port: 80
    targetPort: 80
  - name: https
    port: 443
    targetPort: 80
```

The Load Balancer's annotations are of particular importance. The ports on which to support https traffic are defined by the value of **oci-load-balancer-ssl-ports**. You can declare multiple SSL ports by using a comma-separated list for the annotation's value. For example, you could set the annotation's value to "443, 3000" to support SSL on ports 443 and 3000.

The required TLS secret, **ssl-certificate-secret**, needs to be created in Kubernetes. This example creates and uses a self-signed certificate. However, in a production environment, the most common scenario is to use a public certificate that's been signed by a certificate authority.

The following command creates a self-signed certificate, `tls.crt`, with its corresponding key, `tls.key`:

```
$ openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout tls.key -out tls.crt -subj
"/CN=nginxsvc/O=nginxsvc"
```

Now that you created the certificate, you need to store both it and its key as a secret in Kubernetes. The name of the secret must match the name from the **oci-load-balancer-tls-secret** annotation of the load balancer's definition. Use the following command to create a TLS secret in Kubernetes, whose key and certificate values are set by `--key` and `--cert`, respectively.

```
$ kubectl create secret tls ssl-certificate-secret --key tls.key --cert tls.crt
```

You must create the Kubernetes secret before you can create the service, since the service references the secret in its definition. Create the service using the following command:

```
$ kubectl create -f manifests/demo/nginx-demo-svc-ssl.yaml
```

Watch the service and wait for a public IP address (EXTERNAL-IP) to be assigned to the Nginx service (nginx-service). This is the load balancer IP to use to connect to the service.

```
$ kubectl get svc --watch

NAME            CLUSTER-IP     EXTERNAL-IP      PORT(S)         AGE
nginx-service   192.0.2.1      198.51.100.1     80:30274/TCP    5m
```

The load balancer is now running, which means the service can now be accessed using either http or https, as demonstrated by the following commands:

```
$ curl http://198.51.100.1

$ curl --insecure https://198.51.100.1
```

The "`--insecure`" flag is used to access the service using https due to the use of self-signed certificates in this example. Do not use this flag in a production environment where the public certificate was signed by a certificate authority.

Note: When a cluster is deleted, a load balancer that's dynamically created when a service is created will not be removed. Before deleting a cluster, delete the service, which in turn will result in the cloud provider removing the load balancer. The syntax for this command is:

```
$ kubectl delete svc SERVICE_NAME
```

For example, to delete the service from the previous example, enter:

```
$ kubectl delete svc nginx-service
```

Creating Internal Load Balancers in Public and Private Subnets

When you create a 'custom' cluster, you select an existing VCN that contains the network resources to be used by the new cluster. If you want to use load balancers to control traffic into the VCN, you select existing public or private subnets in that VCN to host the load balancers.

When you create a 'quick cluster', the VCN that's automatically created contains public subnets to host load balancers. If you want to host load balancers in private subnets, you can add private subnets to the VCN later.

An internal load balancer restricts access to a cluster to only applications running in the same VCN as the cluster. You can host internal load balancers in public subnets and private subnets.

To enable a service to use an internal load balancer, include the following annotation in the configuration file:

```
service.beta.kubernetes.io/oci-load-balancer-internal: "true"
```

If you intend to host an internal load balancer in a private subnet, Oracle recommends you select the **Availability Domain-Specific** option when creating the private subnet. To enable a service to use the internal load balancer in that private subnet, include both the following annotations in the configuration file:

```
service.beta.kubernetes.io/oci-load-balancer-internal: "true"

service.beta.kubernetes.io/oci-load-balancer-subnet1: "ocid1.subnet.oc1..aaaaaa....vdfw"
```

where `ocid1.subnet.oc1..aaaaaa....vdfw` is the OCID of the private subnet you created.

For example:

```
apiVersion: v1
kind: Service
metadata:
  name: my-nginx-svc
  labels:
    app: nginx
  annotations:
    service.beta.kubernetes.io/oci-load-balancer-internal: "true"
    service.beta.kubernetes.io/oci-load-balancer-subnet1: "ocid1.subnet.oc1..aaaaaa....vdfw"
spec:
  type: LoadBalancer
  ports:
  - port: 8100
  selector:
    app: nginx
```

Specifying Alternative Load Balancer Shapes

The shape of an Oracle Cloud Infrastructure load balancer specifies its maximum total bandwidth (that is, ingress plus egress). By default, load balancers are created with a shape of 100Mbps. Other shapes are available, including 400Mbps and 8000Mbps. To specify an alternative shape for a load balancer, add an annotation in the metadata section of the manifest file.

For example:

```
apiVersion: v1
kind: Service
metadata:
  name: my-nginx-svc
  labels:
    app: nginx
  annotations:
    service.beta.kubernetes.io/oci-load-balancer-shape: 400Mbps
spec:
  type: LoadBalancer
  ports:
  - port: 80
  selector:
    app: nginx
```

Note: Sufficient load balancer quota must be available in the region for the shape you specify. Enter the following kubectl command to confirm that load balancer creation did not fail due to lack of quota:

```
$ kubectl describe service <service-name>
```

Creating a Persistent Volume Claim

Container storage via a container's root file system is ephemeral, and can disappear upon container deletion and creation. To provide a durable location to store data and prevent it from being lost, you can create and use persistent volumes to store data outside of containers.

You can define and apply a persistent volume claim to your cluster, which in turn creates a persistent volume that's bound to the claim. A claim is a block storage volume in the

underlying IaaS provider that's durable and offers persistent storage, enabling your data to remain intact, regardless of whether the containers that the storage is connected to are terminated.

With Oracle Cloud Infrastructure as the underlying IaaS provider, you can provision persistent volume claims by attaching volumes from the Block Storage service.

A persistent volume claim (PVC) is a request for storage, similar to how a pod requests compute resources. A PVC provides an abstraction layer to underlying storage. For example, an administrator could create a number of static persistent volumes (PVs) that can later be bound to one or more persistent volume claims. This is analogous to an administrator creating cluster nodes to which pods are later assigned. If none of the static persistent volumes match the user's PVC request, the cluster may attempt to dynamically create a PV that matches the PVC request. This example uses the latter approach, and it assumes that the cluster administrator has not created any suitable PVs that match the PVC request—meaning that the PVCs will dynamically create the PVs for this example.

The minimum amount of persistent storage that a PVC can request is 50 gigabytes. If the request is for less than 50 gigabytes, the request is rounded up to 50 gigabytes.

The following YAML defines two PVCs that each request 50 gigabytes of persistent storage (`storage: 50Gi`). You use names of the PVCs (for example, mysqlclaim) when defining which claims to use as the volumes of a deployment.

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: mysqlclaim
spec:
  storageClassName: "oci"
  selector:
    matchLabels:
      failure-domain.beta.kubernetes.io/zone: "US-ASHBURN-AD-1"
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 50Gi
---
apiVersion: v1
```

```
kind: PersistentVolumeClaim
metadata:
  name: wordpressclaim
spec:
  storageClassName: "oci"
  selector:
    matchLabels:
      failure-domain.beta.kubernetes.io/zone: "US-ASHBURN-AD-2"
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 50Gi
```



Note

In the previous example, the PVCs request storage in availability domains in the Ashburn region using `matchLabels:failure-domain.beta.kubernetes.io/zone`. Note that when you specify values for `matchLabels:failure-domain.beta.kubernetes.io/zone`, you must use the shortened versions of availability domain names. For example `US-ASHBURN-AD-1`, `US-ASHBURN-AD-2`. See Availability by Region Name and Region Code for a list of shortened versions of availability domain names.

Enter the following command to create the PVC from the YAML file:

```
$ kubectl create -f https://raw.githubusercontent.com/wercker/oke_examples/master/kubernetes_
examples/persistent_volume_claims.yaml

persistentvolumeclaim "mysqlclaim" created
persistentvolumeclaim "wordpressclaim" created
```

You can verify that the PVCs have been created and bound to persistent volumes by calling `kubectl get pvc`:

```
$ kubectl get pvc

NAME             STATUS    VOLUME                                                                     CAPACITY
ACCESSMODES   STORAGECLASS   AGE
mysqlclaim       Bound     abyhqljrerxpanjto7b5zlxjesy4aedghc5c52f5v43xcrymo77ktdl6ibjq   50Gi       RWO
          oci            4m
wordpressclaim   Bound     abyhqljt3rzldcclootxn7yrfgv36s7rnggcobennjohevykqpitzkinspka   50Gi       RWO
          oci            4m
```

You can use these persistent volumes when creating other objects, such as deployments. For example, the following deployment definition instructs the system to use the mysqlclaim PVC as the mysql-persistent-storage volume, which is mounted by pods hosting the deployment as /var/lib/mysql.

```
#MySQL Deployment
apiVersion: extensions/v1beta1
kind: Deployment
metadata:
  name: mysql
  labels:
    app: mysql
spec:
  replicas: 1
  selector:
    matchLabels:
      app: mysql
  template:
    metadata:
      labels:
        app: mysql
    spec:
      containers:
        - image: mysql:5.6
          name: mysql
          env:
            - name: MYSQL_ROOT_PASSWORD
              valueFrom:
                secretKeyRef:
                  name: mysql
```

```
                key: password
        ports:
          - containerPort: 3306
            name: mysql
        volumeMounts:
          - name: mysql-persistent-storage
            mountPath: /var/lib/mysql
    volumes:
      - name: mysql-persistent-storage
        persistentVolumeClaim:
          claimName: mysqlclaim
```

Adding OCI Service Broker for Kubernetes to Clusters

Service brokers offer a catalog of backing services to workloads running on cloud native platforms. The Open Service Broker API is a commonly-used standard for interactions between service brokers and platforms. The Open Service Broker API specification describes a simple set of API endpoints that platforms use to provision, gain access to, and manage service offerings. For more information about the Open Service Broker API, see resources available online including those at openservicebrokerapi.org.

OCI Service Broker for Kubernetes is an implementation of the Open Service Broker API. OCI Service Broker for Kubernetes is specifically for interacting with Oracle Cloud Infrastructure services from Kubernetes clusters. It includes three service broker adapters to bind to the following Oracle Cloud Infrastructure services:

- Object Storage
- Autonomous Transaction Processing
- Autonomous Data Warehouse

You can add OCI Service Broker for Kubernetes to clusters you've created with Oracle Cloud Infrastructure Container Engine for Kubernetes to interact with the Oracle Cloud Infrastructure services listed above. Having added OCI Service Broker for Kubernetes to a cluster, you don't have to manually provision and de-provision the Oracle Cloud Infrastructure services each time you deploy or un-deploy an application on the cluster. Instead, you interact with the Oracle Cloud Infrastructure services by using kubectl to call the Open Service Broker APIs implemented by OCI Service Broker for Kubernetes .

OCI Service Broker for Kubernetes is available as a Helm chart, a Docker container, and as source code from Github.

For more information about OCI Service Broker for Kubernetes, see the OCI Service Broker for Kubernetes documentation in the Github repository.

Adding OCI Service Broker for Kubernetes to a Cluster

To add OCI Service Broker for Kubernetes to a cluster, follow the detailed instructions in the Github repository.

For convenience, here's a high-level summary of the steps involved:

1. Install OCI Service Broker for Kubernetes. During this step, you will typically:
   - Install the Service Catalog.
   - Install the svcat tool.
   - Deploy OCI Service Broker for Kubernetes.
   - Grant RBAC permissions and roles.
   - Register OCI Service Broker for Kubernetes.

   For more information about installation, see the OCI Service Broker for Kubernetes documentation in the Github repository.
2. Secure OCI Service Broker for Kubernetes. During this step, you will typically:
   - Restrict access to Service Catalog resources using RBAC permissions and roles.
   - Configure TLS for OCI Service Broker for Kubernetes.
   - Set up an Oracle Cloud Infrastructure user for use by OCI Service Broker for Kubernetes.
   - Set up appropriate policies to control access to resources (according to the Oracle Cloud Infrastructure services to be used).
   - Limit access to the OCI Service Broker for Kubernetes endpoint using NetworkPolicy.

- Stand up an etcd cluster for Service Catalog and OCI Service Broker for Kubernetes.
- Protect sensitive values by creating secrets.

The security configuration to choose will depend on your particular requirements. For more information, see the OCI Service Broker for Kubernetes documentation in the Github repository.

3. Provision and bind to the required Oracle Cloud Infrastructure services. During this step, you will typically:
   - Provide service provision request parameters.
   - Provide service binding request parameters.
   - Provide service binding response credentials.

   The details to provide will depend on the Oracle Cloud Infrastructure service to bind to. For more information, see the OCI Service Broker for Kubernetes documentation in the Github repository.

Example: Setting Up an Ingress Controller on a Cluster

You can set up different open source ingress controllers on clusters you have created with Container Engine for Kubernetes.

This topic explains how to set up an example ingress controller along with corresponding access control on an existing cluster. Having set up the ingress controller, this topic describes how to use the ingress controller with an example hello-world backend, and how to verify the ingress controller is working as expected.

Example Components

The example includes an ingress controller and a hello-world backend.

Ingress Controller Components

The ingress controller comprises:

- An ingress controller deployment called `nginx-ingress-controller`. The deployment deploys an image that contains the binary for the ingress controller and Nginx. The binary manipulates and reloads the `/etc/nginx/nginx.conf` configuration file when an ingress is created in Kubernetes. Nginx upstreams point to services that match specified selectors.
- An ingress controller service called `ingress-nginx`. The service exposes the ingress controller deployment as a LoadBalancer type service. Because Container Engine for Kubernetes uses an Oracle Cloud Infrastructure integration/cloud-provider, a load balancer will be dynamically created with the correct nodes configured as a backend set.

Backend Components

The hello-world backend comprises:

- A backend deployment called `docker-hello-world`. The deployment handles default routes for health checks and 404 responses. This is done by using a stock hello-world image that serves the minimum required routes for a default backend.
- A backend service called `docker-hello-world-svc`.The service exposes the backend deployment for consumption by the ingress controller deployment.

Setting Up the Example Ingress Controller

In this section, you create the access rules for ingress. You then create the example ingress controller components, and confirm they are running.

Creating the Access Rules for the Ingress Controller

1. If you haven't already done so, download the cluster's kubeconfig configuration file and set the KUBECONFIG environment variable to point to the file. See Downloading a kubeconfig File to Enable Cluster Access.
2. In a terminal window, grant the Kubernetes RBAC cluster-admin clusterrole to the user by entering:

   ```
   $ kubectl create clusterrolebinding <my-cluster-admin-binding> --clusterrole=cluster-admin --
   user=<user_OCID>
   ```

   where:

   - `<my-cluster-admin-binding>` is a string of your choice to be used as the name for the binding between the user and the Kubernetes RBAC cluster-admin clusterrole. For example, `jdoe_clst_adm`
   - `<user_OCID>` is the user's OCID (obtained from the Console). For example, `ocid1.user.oc1..aaaaa...zutq` (abbreviated for readability).

   For example:

   ```
   $ kubectl create clusterrolebinding jdoe_clst_adm --clusterrole=cluster-admin --
   user=ocid1.user.oc1..aaaaa...zutq
   ```

Creating the Service Account, and the Ingress Controller

1. Run the following command to create the `nginx-ingress-controller` ingress controller deployment, along with the Kubernetes RBAC roles and bindings:

   ```
   $ kubectl apply -f https://raw.githubusercontent.com/kubernetes/ingress-
   nginx/master/deploy/mandatory.yaml
   ```

2. Create and save the file `cloud-generic.yaml` containing the following code to define the `ingress-nginx` ingress controller service as a load balancer service:

   ```
   kind: Service
   apiVersion: v1
   metadata:
     name: ingress-nginx
     namespace: ingress-nginx
     labels:
   ```

```
    app.kubernetes.io/name: ingress-nginx
    app.kubernetes.io/part-of: ingress-nginx
spec:
  type: LoadBalancer
  selector:
    app.kubernetes.io/name: ingress-nginx
    app.kubernetes.io/part-of: ingress-nginx
  ports:
    - name: http
      port: 80
      targetPort: http
    - name: https
      port: 443
      targetPort: https
```

3. Using the file you just saved, create the `ingress-nginx` ingress controller service by running the following command:

```
$ kubectl apply -f cloud-generic.yaml
```

Verifying the `ingress-nginx` Ingress Controller Service is Running as a Load Balancer Service

1. View the list of running services:

```
$ kubectl get svc -n ingress-nginx

NAME            TYPE           CLUSTER-IP     EXTERNAL-IP    PORT(S)                      AGE
ingress-nginx   LoadBalancer   10.96.229.38   <pending>      80:30756/TCP,443:30118/TCP   1h
```

   The EXTERNAL-IP for the `ingress-nginx` ingress controller service is shown as `<pending>` until the load balancer has been fully created in Oracle Cloud Infrastructure.

2. Repeat the `kubectl get svc` command until an EXTERNAL-IP is shown for the `ingress-nginx` ingress controller service:

```
$ kubectl get svc -n ingress-nginx

NAME            TYPE           CLUSTER-IP     EXTERNAL-IP       PORT(S)                      AGE
ingress-nginx   LoadBalancer   10.96.229.38   129.146.214.219   80:30756/TCP,443:30118/TCP   1h
```

Creating the TLS Secret

A TLS secret is used for SSL termination on the ingress controller. To generate the secret for this example, a self-signed certificate is used. While this is okay for testing, for production, use a certificate signed by a Certificate Authority.

```
$ openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout tls.key -out tls.crt -subj
"/CN=nginxsvc/O=nginxsvc"

$ kubectl create secret tls tls-secret --key tls.key --cert tls.crt
```



Note

Under Windows, you may need to replace `"/CN=nginxsvc/O=nginxsvc"` with `"//CN=nginxsvc\O=nginxsvc"` . For example, this is necessary if you run the `openssl` command from a Git Bash shell.

Setting Up the Example Backend

In this section, you define a hello-world backend service and deployment.

Creating the docker-hello-world Service Definition

1. Create the file `hello-world-ingress.yaml` containing the following code. This code uses a publicly available hello-world image from Docker Hub. You can substitute another image of your choice that can be run in a similar manner.

```
apiVersion: apps/v1beta1
kind: Deployment
metadata:
  name: docker-hello-world
  labels:
    app: docker-hello-world
```

```
spec:
  replicas: 3
  template:
    metadata:
      labels:
        app: docker-hello-world
    spec:
      containers:
      - name: docker-hello-world
        image: scottsbaldwin/docker-hello-world:latest
        ports:
        - containerPort: 80
---
apiVersion: v1
kind: Service
metadata:
  name: docker-hello-world-svc
spec:
  selector:
    app: docker-hello-world
  ports:
    - port: 8088
      targetPort: 80
  type: ClusterIP
```

Note the docker-hello-world service's type is ClusterIP, rather than LoadBalancer, because this service will be proxied by the `ingress-nginx` ingress controller service. The docker-hello-world service does not need public access directly to it. Instead, the public access will be routed from the load balancer to the ingress controller, and from the ingress controller to the upstream service.

2. Create the new hello-world deployment and service on nodes in the cluster by running the following command:

```
$ kubectl create -f hello-world-ingress.yaml
```

Using the Example Ingress Controller to Access the Example Backend

In this section you create an ingress to access the backend using the ingress controller.

Creating the Ingress Resource

1. Create the file `ingress.yaml` and populate it with this code:

```
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: hello-world-ing
  annotations:
    kubernetes.io/ingress.class: "nginx"
spec:
  tls:
  - secretName: tls-secret
  rules:
  - http:
      paths:
      - backend:
          serviceName: docker-hello-world-svc
          servicePort: 8088
```

2. Create the resource:

```
$ kubectl create -f ingress.yaml
```

Verifying that the Example Components are Working as Expected

In this section, you confirm that all of the example components have been successfully created and are operating as expected. The `docker-hello-world-svc` service should be running as a ClusterIP service, and the `ingress-nginx` service should be running as a LoadBalancer service. Requests sent to the ingress controller should be routed to nodes in the cluster.

Obtaining the External IP Address of the Load Balancer

To confirm the `ingress-nginx` service is running as a LoadBalancer service, obtain its external IP address:

```
$ kubectl get svc --all-namespaces

NAMESPACE       NAME                    TYPE           CLUSTER-IP      EXTERNAL-IP      PORT(S)
             AGE
```

```
default          docker-hello-world-svc   ClusterIP      10.96.83.247    <none>           8088/TCP
             16s
default          kubernetes               ClusterIP      10.96.0.1       <none>           443/TCP
             1h
ingress-nginx    ingress-nginx            LoadBalancer   10.96.229.38    129.146.214.219
80:30756/TCP,443:30118/TCP   5m
kube-system      kube-dns                 ClusterIP      10.96.5.5       <none>           53/UDP,53/TCP
             1h
kube-system      kubernetes-dashboard     ClusterIP      10.96.208.64    <none>           443/TCP
             1h
kube-system      tiller-deploy            ClusterIP      10.96.28.102    <none>           44134/TCP
             1h
```

Sending cURL Requests to the Load Balancer

1. Use the external IP address of the `ingress-nginx` service (for example, 129.146.214.219) to curl an http request:

   ```
   $ curl -I http://129.146.214.219

   HTTP/1.1 301 Moved Permanently
   Via: 1.1 10.68.69.10 (McAfee Web Gateway 7.6.2.10.0.23236)
   Date: Thu, 07 Sep 2017 15:20:16 GMT
   Server: nginx/1.13.2
   Location: https://129.146.214.219/
   Content-Type: text/html
   Content-Length: 185
   Proxy-Connection: Keep-Alive
   Strict-Transport-Security: max-age=15724800; includeSubDomains;
   ```

   The output shows a 301 redirect and a Location header that suggest that http traffic is being redirected to https.

2. Either cURL against the https url or add the `-L` option to automatically follow the location header. The `-k` option instructs cURL to not verify the SSL certificates.

   ```
   $ curl -ikL http://129.146.214.219

   HTTP/1.1 301 Moved Permanently
   Via: 1.1 10.68.69.10 (McAfee Web Gateway 7.6.2.10.0.23236)
   Date: Thu, 07 Sep 2017 15:22:29 GMT
   ```

```
Server: nginx/1.13.2
Location: https://129.146.214.219/
Content-Type: text/html
Content-Length: 185
Proxy-Connection: Keep-Alive
Strict-Transport-Security: max-age=15724800; includeSubDomains;

HTTP/1.0 200 Connection established

HTTP/1.1 200 OK
Server: nginx/1.13.2
Date: Thu, 07 Sep 2017 15:22:30 GMT
Content-Type: text/html
Content-Length: 71
Connection: keep-alive
Last-Modified: Thu, 07 Sep 2017 15:17:24 GMT
ETag: "59b16304-47"
Accept-Ranges: bytes
Strict-Transport-Security: max-age=15724800; includeSubDomains;

<h1>Hello webhook world from: docker-hello-world-1732906117-0ztkm</h1>
```

The last line of the output shows the HTML that is returned from the pod whose hostname is `docker-hello-world-1732906117-0ztkm`.

3. Issue the cURL request several times to see the hostname in the HTML output change, demonstrating that load balancing is occurring:

```
$ curl -k https://129.146.214.219

<h1>Hello webhook world from: docker-hello-world-1732906117-6115l</h1>

$ curl -k https://129.146.214.219

<h1>Hello webhook world from: docker-hello-world-1732906117-7r89v</h1>

$ curl -k https://129.146.214.219

<h1>Hello webhook world from: docker-hello-world-1732906117-0ztkm</h1>
```

Inspecting nginx.conf

The `nginx-ingress-controller` ingress controller deployment manipulates the `nginx.conf` file in the pod within which it is running.

1. Find the name of the pod running the `nginx-ingress-controller` ingress controller deployment and use it with a `kubectl exec` command to show the contents of `nginx.conf`.

```
$ kubectl get po -n ingress-nginx

NAME                                        READY     STATUS    RESTARTS   AGE
nginx-ingress-controller-110676328-h86xg   1/1       Running   0          1h

$ kubectl exec -n ingress-nginx -it nginx-ingress-controller-110676328-h86xg -- cat
/etc/nginx/nginx.conf
```

2. Look for `proxy_pass` in the output. There will be one for the default backend and another that looks similar to:

```
proxy_pass http://upstream_balancer;
```

   This shows that Nginx is proxying requests to an upstream called `upstream_balancer`.

3. Locate the upstream definition in the output. It will look similar to:

```
upstream upstream_balancer {
                server 0.0.0.1:1234; # placeholder

                balancer_by_lua_block {
                        tcp_udp_balancer.balance()
                }
        }
```

   The upstream is proxying via Lua.

CHAPTER 8 Data Transfer

This chapter explains how to migrate data to Oracle Cloud Infrastructure using Data Transfer Disk and Data Transfer Appliance.

Overview of Data Transfer Service

Oracle offers offline data transfer solutions that let you migrate data to Oracle Cloud Infrastructure. Moving data over the public internet is not always feasible due to high network costs, unreliable network connectivity, long transfer times, and security concerns. Our transfer solutions address these pain points, are easy to use, and provide significantly faster data upload compared to over-the-wire data transfer.

Data transfer is currently supported in us-ashburn-1, us-phoenix-1, eu-frankfurt-1, and uk-london-1 regions.



Note

To simplify this Data Transfer documentation, we generically refer to Object Storage to mean that you can transfer data into a bucket in either the Object Storage tier or Archive Storage tier.

DATA TRANSFER DISK

You send your data as files on encrypted commodity hard disk drives to an Oracle transfer site. Operators at the Oracle transfer site upload the files into your designated Object Storage bucket in your tenancy.

This transfer solution requires you to source and purchase the disks used to transfer data to Oracle Cloud Infrastructure. The disks are shipped back to you after the data is successfully uploaded.

See Data Transfer Disk for details.

DATA TRANSFER APPLIANCE

You send your data as files on secure, high-capacity, Oracle-supplied storage appliances to an Oracle transfer site. Operators at the Oracle transfer site upload the data into your designated Object Storage bucket in your tenancy.

This solution supports data transfer when you are migrating a large volume of data and when using disks is not a practical alternative. You do not need to write any code or purchase any hardware—Oracle supplies the transfer appliance and all of the software required to manage the transfer.

See Data Transfer Appliance for details.

Limits on Data Transfer Service Resources

When you sign up for Oracle Cloud Infrastructure, a set of service limits are configured for your tenancy. The service limit is the quota or allowance set on a resource. Limits for Data Transfer are initially set to zero. Verify that your service limits are set appropriately before you begin the data transfer process.

See Service Limits for a list of applicable limits and instructions for requesting a limit increase.

Data Transfer Disk

Data Transfer Disk is one of Oracle's offline data transfer solutions that lets you migrate data to Oracle Cloud Infrastructure. You send your data as files on encrypted disks to an Oracle transfer site. Operators at the Oracle transfer site upload the files into the designated Object Storage bucket in your tenancy. You are then free to move the uploaded data to other Oracle Cloud Infrastructure services as needed.

Data Transfer Disk Concepts

The following concepts are essential to understanding Data Transfer Service.

TRANSFER JOB

A transfer job is the logical representation of a data migration to Oracle Cloud Infrastructure. A transfer job consists of one or more transfer packages that each contain one or more transfer disks.

TRANSFER DISK

A transfer disk is an HDD that is specially prepared to copy and upload data to Oracle Cloud Infrastructure. You copy your data to one or more of these disks and ship the disks in a parcel to Oracle to upload your data.

The following transfer disks are supported:

- SATA II/III 2.5" or 3.5" HDDs
- External USB 2.0/3.0 HDDs

DATA TRANSFER UTILITY

The Data Transfer Utility is the command-line software that Oracle provides for you to prepare transfer disks for your data and for shipment to Oracle. In addition, you can use this software to manage transfer jobs and packages.

HOST

The computer at your site on which you download the Data Transfer Utility to perform Data Transfer Service tasks.

TRANSFER PACKAGE

A transfer package is the logical representation of the parcel containing the transfer disks that you ship to Oracle to upload to Oracle Cloud Infrastructure.

BUCKET

The logical container in Oracle Cloud Infrastructure Object Storage where Oracle operators upload your data. A bucket is associated with a single compartment in your

tenancy that has policies that determine what actions a user can perform on a bucket and on all the objects in the bucket.

DATA TRANSFER ADMINISTRATOR

A new or existing IAM user that has the authorization and permissions to create and manage transfer jobs. See Preparing for Data Transfer.

DATA TRANSFER UPLOAD USER

A temporary IAM user that grants Oracle personnel the authorization and permissions to upload the data from your transfer disks to your designated Oracle Cloud Infrastructure Object Storage bucket. Delete this temporary user after your data is uploaded to Oracle Cloud Infrastructure. See Preparing for Data Transfer.

Task Flow for Data Transfer Disk

Here is a high-level overview of the tasks involved in transferring data to Oracle Cloud Infrastructure using Data Transfer Disk.

Performing prerequisite tasks in preparation for transfer data

An Oracle Cloud Infrastructure administrator must perform prerequisite tasks in preparation for data transfer. These tasks are covered in greater detail in Preparing for Data Transfer.

1. Create or designate a bucket in your tenancy where Oracle is to upload your data.
2. Create or use an existing IAM group for data transfer administrators with the authorization and permissions to create and manage transfer jobs and manage objects in Oracle Cloud Infrastructure Object Storage.
3. Create or use an existing IAM data transfer administrator user and add that user to the data transfer administrators group.
4. Create or use an existing IAM group for data transfer upload users with the authorization and permissions to upload data to Oracle Cloud Infrastructure Object Storage.
5. Create a temporary IAM data transfer upload user and add that user to data transfer upload user group.

6. Write the authorization policies to allow the data transfer administrator and upload user groups to perform the required data transfer tasks.



Note

Important Security Consideration

For security reasons, we recommend that you create a unique IAM data transfer upload user for each transfer job and then delete that user after your data is uploaded to Oracle Cloud Infrastructure.

A data transfer administrator performs the remaining tasks. These tasks are covered in detail in Managing Disk Data Transfers.

Preparing for and copying your data

1. Create a transfer job.
2. Attach HDDs to your host machine.
3. Create transfer disks.
4. Copy your data to the transfer disks.

Finalizing the transfer disks in preparation for shipment

1. Generate a manifest for each transfer disk.
2. Generate the "dry run" report for each transfer disk
3. Lock each transfer disk.

Preparing and shipping the package

1. Create one or more transfer packages.
2. Attach the transfer disks to the transfer packages.
3. Get the shipping address for the transfer packages.

4. Package the transfer disks into a box, and ship the box to Oracle using an approved shipping vendor.
5. Update the transfer package with tracking information.

Secure Disk Data Transfer to Oracle Cloud Infrastructure

This section highlights the security details of the Data Transfer Service process.

- The Data Transfer Utility uses the standard Linux dm-crypt and LUKS utilities to encrypt block devices.
- The dm-crypt software generates a master AES-256 bit encryption key that is used for all data written to or read from the disk. That key is protected by an encryption passphrase that the user must know to access the encrypted data.
- When the data transfer administrator uses the Data Transfer Utility to create disks, Oracle Cloud Infrastructure creates a strong encryption passphrase that is displayed to the user and passed to dm-crypt. The passphrase is displayed to standard output only once and cannot be retrieved again. Copy this passphrase to a durable, secure location for future reference.
- For additional security, you can also encrypt your own data with you own encryption keys. Before copying your data to the transfer disk, you can encrypt your data with a tool and encryption key of your choosing. After the data has been uploaded, you would need to use the same tool and encryption key to access the data.
- All network communication between the Data Transfer Utility and Oracle Cloud Infrastructure is encrypted in-transit using Transport Layer Security (TLS).
- After copying your data to a transfer disk, generate a manifest file using the Data Transfer Utility. The manifest contains an index of all of the copied files and generated data integrity hashes. The Data Transfer Utility copies the `config_upload_user` configuration file and referenced IAM credentials to the encrypted transfer disk. This configuration file describes the temporary IAM data transfer upload user. Oracle uses the credentials and entries defined in the `config_upload_user` file when processing the transfer disk and uploading files to Oracle Cloud Infrastructure Object Storage.



Note

Data Transfer Service Does Not Support Passphrases on Private Keys

While we recommend encrypting a private key with a passphrase when generating API signing keys, Data Transfer does not support passphrases on the key file required for the `config_upload_user`. If you use a passphrase, Oracle personnel cannot upload your data.

Oracle cannot upload data from a transfer disk without the correct credentials defined in this configuration file. See Installing and Configuring the Data Transfer Utility for more information about the required configuration files.

- When you disconnect or lock a transfer disk using the Data Transfer Utility, the original encryption passphrase is required to once again access the disk. If the encryption passphrase is not known or lost, you cannot access the data on the transfer disk. To reuse a transfer disk, you must reformat the disk. Reformatting a disk removes all of the data.
- Oracle retrieves the encryption passphrase for a transfer disk from Oracle Cloud Infrastructure. Oracle uses the passphrase to decrypt, mount the transfer disk, and upload the data to the designated bucket in the tenancy.
- After processing a transfer package, Oracle returns all transfer disks attached to the transfer package using the return shipping label you provide.
- To protect your data, we make the data on the disk unrecoverable before shipping the transfer disks back to you. To comply with customs regulations, we wipe the disks completely before shipping the transfer disks back to international shipping addresses.

Ways to Manage Disk Data Transfers

We provide two ways to manage Data Transfer Services:

- The Data Transfer Utility is a full-featured command-line tool. For more information and installation instructions, see Installing and Configuring the Data Transfer Utility.
- The Console is an easy-to-use, partial-featured browser-based interface. For more information, see Signing In to the Console.



Note

You can perform many data transfer tasks using either the Console or the Data Transfer Utility. However, there are some tasks you can **only** perform using the Data Transfer Utility (for example, creating and locking transfer disks). Managing Disk Data Transfers describes the management tasks in detail and guides you to the appropriate management interface to use for each task.

Data Transfer Appliance

Data Transfer Appliance is one of Oracle's offline data transfer solutions that lets you migrate petabyte-scale datasets to Oracle Cloud Infrastructure. You send your data as files on one or more secure, high-capacity, Oracle-supplied storage appliances to an Oracle transfer site. Operators at the Oracle transfer site upload the files into the designated Object Storage bucket in your tenancy. You are then free to move the uploaded data to other Oracle Cloud Infrastructure services as needed.

Data Transfer Appliance Concepts

The following concepts are essential to understanding Data Transfer Appliance.

TRANSFER JOB

A transfer job is the logical representation of a data migration to Oracle Cloud Infrastructure. A transfer job is associated with one or more transfer appliances.

TRANSFER APPLIANCE

A transfer appliance is high storage capacity device that is specially prepared to copy and upload data to Oracle Cloud Infrastructure. You request a transfer appliance from Oracle, copy your data to the appliance, and then ship the appliance back to Oracle to upload your data.

DATA TRANSFER UTILITY

The Data Transfer Utility is the command-line software that Oracle provides for you to prepare transfer appliances for your data and for shipment to Oracle. In addition, you can use this software to manage transfer jobs and obtain status information about your appliances.

HOST

The computer at your site on which you download the Data Transfer Utility to perform Data Transfer Appliance tasks.

BUCKET

The logical container in Oracle Cloud Infrastructure Object Storage where Oracle operators upload your data. A bucket is associated with a single compartment in your tenancy that has policies that determine what actions a user can perform on a bucket and on all the objects in the bucket.

DATA TRANSFER ADMINISTRATOR

A new or existing IAM user that has the authorization and permissions to create and manage transfer jobs. See Preparing for Data Transfer.

DATA TRANSFER UPLOAD USER

A temporary IAM user that grants Oracle personnel the authorization and permissions to upload the data from your transfer devices to your designated Oracle Cloud Infrastructure Object Storage bucket. Delete this temporary user after your data is uploaded to Oracle Cloud Infrastructure. See Preparing for Data Transfer.

DATA TRANSFER APPLIANCE MANAGEMENT SERVICE

Software running on the transfer appliance that provides management functions. Users interact with this service though the Data Transfer Utility.

Transfer Appliance Specifications

You use NFSv3 to copy your data onto the Oracle data transfer appliance. Here are some details about the appliance:

Item Description	Specification
Storage Capacity	150 TB of protected usable space
Network Interfaces	- 10 GbE - RJ45 - 10 GbE - SFP+ You are responsible for providing all network cables. If you want to use SFP+, your transceivers must be compatible with Intel X520 NICs.
Provided Cables	- NEMA 5-15 type B to C13 - C13-14 power - USB-DB9 serial

Item Description	Specification
Environmental	- Operational temperature: 10–35°C - Operational relative humidity: 8–90% non-condensing - Acoustics: < 75 dB @ 23° C - Operational altitude: -300–3048 m (10,000 ft)
Power	- Consumption: 554 W - Voltage: 100–240 VAC - Frequency: 47–63 Hz - Conversion efficiency: 89%
Weight	- Unit: 38 lbs (17.2365 kg) - Unit + Transit Case: 64 lbs (29.0299 kg)
Height	3.5" (2U)
Width	17"
Depth	24"

Task Flow for Data Transfer Appliance

Here is a high-level overview of the tasks involved in transferring data to Oracle Cloud Infrastructure using Data Transfer Appliance.

Performing prerequisite tasks in preparation for data transfer

An Oracle Cloud Infrastructure administrator must perform prerequisite tasks in preparation for data transfer. These tasks and are covered in detail in Preparing for Data Transfer.

1. Create or designate a bucket in your tenancy where Oracle is to upload your data.
2. Create or use an existing IAM group for data transfer administrators with the authorization and permissions to create and manage transfer jobs and manage objects in Oracle Cloud Infrastructure Object Storage.
3. Create or use an existing IAM data transfer administrator user and add that user to the data transfer administrators group.
4. Create or use an existing IAM group for data transfer upload users with the authorization and permissions to upload data to Oracle Cloud Infrastructure Object Storage.
5. Create a temporary IAM data transfer upload user and add that user to data transfer upload user group.
6. Write the authorization policies to allow the data transfer administrator and upload user groups to perform the required data transfer tasks.



Note

Important Security Consideration

For security reasons, we recommend creating a unique IAM data transfer upload user for each transfer job and then deleting that user after your data is uploaded to Oracle Cloud Infrastructure.

A data transfer administrator performs the remaining tasks. These tasks are covered in detail in Managing Appliance Data Transfers.

Preparing for and copying your data

1. Create a transfer job.
2. Request a transfer appliance.
3. Monitor your transfer appliance request.
4. Set up your host machine.

5. Unpack and prepare your transfer appliance.
6. Configure networking on your transfer appliance.
7. Write data to your transfer appliance.

Preparing your appliances for shipment

1. Finalize your transfer appliance.
2. Package and ship the transfer appliance to Oracle.

Post shipment tasks

1. Optionally, cancel a transfer appliance if you don't want Oracle to upload your data.
2. Monitor your transfer appliance return shipment.
3. Review transfer appliance log files.
4. Close the transfer job.

Secure Appliance Data Transfer to Oracle Cloud Infrastructure

This section highlights the security details of the Data Transfer Appliance process.

- Appliances are shipped from Oracle to you with a tamper-evident security tie on the transit case. A second tamper-evident security tie is included in the appliance transit case for you to secure the case when you ship the case back to Oracle. The number on the physical security ties must match the numbers logged by Oracle in the transfer appliance details.
- When you configure the transfer appliance for the first time:
    - The transfer appliance generates a master AES-256 bit encryption key that is used for all data written to or read from the device. The encryption key never leaves the device.
    - The encryption key is protected by an encryption passphrase that you must know to access the encrypted data. The Data Transfer Utility securely fetches a provided encryption passphrase from Oracle Cloud Infrastructure and registers that passphrase on the transfer appliance.

- All data is encrypted as the data is copied to an Oracle transfer appliance.
- For additional security, you can also encrypt your own data with you own encryption keys. Before copying your data to the transfer appliance, you can encrypt your data with a tool and encryption key of your choosing. After the data has been uploaded, you would need to use the same tool and encryption key to access the data.
- All network communication between the Data Transfer Utility and Oracle Cloud Infrastructure is encrypted in-transit using Transport Layer Security (TLS).
- After copying your data to a transfer appliance, the Data Transfer Utility generates a manifest file. The manifest contains an index of all of the copied files and generated data integrity hashes. The Data Transfer Utility also encrypts and copies the `config_upload_user` configuration file to the transfer appliance. This configuration file describes the temporary IAM data transfer upload user. Oracle uses the credentials and entries defined in the `config_upload_user` file when processing the transfer appliance and uploading files to Oracle Cloud Infrastructure Object Storage.

  Note

  Data Transfer Service Does Not Support Passphrases on Private Keys

  While we recommend encrypting a private key with a passphrase when generating API signing keys, Data Transfer does not support passphrases on the key file required for the `config_upload_user`. If you use a passphrase, Oracle personnel cannot upload your data.

  Oracle cannot upload data from a transfer appliance without the correct credentials defined in this configuration file. See Installing and Configuring the Data Transfer Utility for more information about the required configuration files.
- Oracle erases all of your data from the transfer appliance after it has been processed. The erasure process follows the NIST 800-88 standards.

Ways to Manage Appliance Data Transfers

We provide two ways to manage data transfer appliances:

- The Data Transfer Utility is a full-featured command-line tool. For more information and installation instructions, see Installing and Configuring the Data Transfer Utility.
- The Console is an easy-to-use, partial-featured browser-based interface. For more information, see Signing In to the Console.



Note

You can perform many data transfer tasks using either the Console or the Data Transfer Utility. However, there are some tasks you can **only** perform using the Data Transfer Utility. Managing Appliance Data Transfers describes the management tasks in detail and guides you to the appropriate management interface to use for each task.

Preparing for Data Transfer

An Oracle Cloud Infrastructure administrator must perform prerequisite tasks in preparation for data transfer. If you are new to Oracle Cloud Infrastructure, we recommend that you read Setting Up Your Tenancy.

Creating the Required IAM Users, Groups, and Policies

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization.

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have

permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

Access to resources is provided to groups using policies and then inherited by the users that are assigned to those groups. Data transfer requires the creation of two distinct groups:

- Data transfer *administrators* who can create and manage transfer jobs.
- Data transfer *upload users* who can upload data to Object Storage. For your data security, the permissions for upload users allow Oracle personnel to upload standard and multi-part objects on your behalf and inspect bucket and object metadata. The permissions do not allow Oracle personnel to inspect the actual data.

For details on creating groups, see Managing Groups.

An administrator creates these groups with the following policies:

- The data transfer administrator group requires an authorization policy that includes the following:

```
Allow group <group_name> to manage data-transfer-jobs in compartment <compartment_name>
```

```
Allow group <group_name> to manage buckets in compartment <compartment_name>
```

```
Allow group <group_name> to manage objects in compartment <compartment_name>
```

  Alternatively, you can consolidate the `manage buckets` and `manage objects` policies into the following:

```
Allow group <group_name> to manage object-family in compartment <compartment_name>
```

- The data transfer upload user group requires an authorization policy that includes the following:

```
Allow group <group_name> to manage buckets in compartment <compartment_name> where all {
request.permission='BUCKET_READ' }
```

```
Allow group <group_name> to manage objects in compartment <compartment_name> where any {
request.permission='OBJECT_CREATE' , request.permission='OBJECT_OVERWRITE' ,
request.permission='OBJECT_INSPECT' }
```



Important

For security reasons, we recommend that you create a unique IAM data transfer upload user for each transfer job and then delete that user once your data is uploaded to Oracle Cloud Infrastructure.

The Oracle Cloud Infrastructure administrator then adds a user to each of the data transfer groups created. For details on creating users, see Managing Users.

Creating the Required Object Storage Bucket

The Object Storage service is used to upload your data to Oracle Cloud Infrastructure. Object Storage stores objects in a container called a bucket within a compartment in your tenancy. For details on creating the bucket to store uploaded data, see Managing Buckets.

What's Next

Installing the Data Transfer Utility

The Data Transfer Utility is a command-line utility that you install on your host machine. The utility lets you perform all data transfer-related task. This utility is required to perform certain data transfer tasks that cannot be performed using the Console. See Installing and Configuring the Data Transfer Utility for detailed installation and configuration instructions.

Performing Solution-Specific Tasks

After completing the prerequisite tasks, perform the data transfer-related tasks related to the transfer solution you are using:

- If you are using disks to transfer data, see Managing Disk Data Transfers.
- If you are using appliances to transfer data, see Managing Appliance Data Transfers.

Installing and Configuring the Data Transfer Utility

This topic describes how to install and configure the Data Transfer Utility. In addition, this topic describes the syntax for the Data Transfer Utility commands.

- **Services supported:**
  - Data Transfer Disk
  - Data Transfer Appliance
- **Downloading:**
  - Download the .deb file to install on Debian or Ubuntu
  - Download the .rpm file to install on Oracle Linux or Red Hat Linux

Prerequisites

To install and use the Data Transfer Utility, you need the following:

- An Oracle Cloud Infrastructure account.
- Required Oracle Cloud Infrastructure users and groups with the required IAM policies. See Preparing for Data Transfer for details.
- A host machine with the following installed:
  - Oracle Linux 6 or greater, Ubuntu 14.04 or greater, or SUSE 11 or greater
  - Java 1.8 or Java 1.11
  - hdparm 9.0 or later
- For HDD data transfers, the host machine needs the following:
  - Cryptsetup 1.2.0 or greater
- For appliance data transfers, the host machine needs the following:
  - Serial console terminal emulator software. We recommend using the following terminal emulator software:

        - PuTTY for Windows
        - ZOC for OS X
        - PuTTY or Minicom for Linux
    - Terminal emulator software settings as follows:
        - Baud Rate: 115200
        - Emulation: VT102
        - Handshaking: Disabled/off
        - RTS/DTS: Disabled/off
    - The host set up as an NFS client:
        - For Debian or Ubuntu, install the `nfs-common` package.
        - For Oracle Linux or Red Hat Linux, install the `nfs-utils` package.

You might need to set up an HTTP proxy environment to access to the public Internet for the Data Transfer Utility to communicate with the Data Transfer Appliance Management Service and to the transfer appliance over a local network connection. If your environment requires Internet-aware applications to use network proxies, configure the Data Transfer Utility to use your environment's network proxies by setting the standard Linux environment variables on your host machine.

Assume that your organization has a corporate Internet proxy at `http://www-proxy.myorg.com` and that the proxy is an HTTP address at port `80`. You would set the following environment variable:

```
export HTTPS_PROXY=http://www-proxy.myorg.com:80
```

If you configured a proxy on the host machine and the transfer appliance is directly connected to that host, the host machine tries unsuccessfully to communicate with the transfer appliance using a proxy. You must set a `no_proxy` environment variable for the transfer appliance. For example, if the appliance is on a local network at `10.0.0.1`, you would set the following environment variable:

```
export NO_PROXY=10.0.0.1
```

Installing the Data Transfer Utility

Download and install the Data Transfer Utility installer that corresponds to your host machine's operating system.

To install the Data Transfer Utility on Debian or Ubuntu

1. If you have not done so already, download the .deb file.
2. Issue the `apt install` command as the `root` user that has write permissions to the `/opt` directory.

   ```
   sudo apt install ./dts-X.Y.Z.x86_64.deb
   ```

   `X.Y.Z` represents the version numbers that match the installer you downloaded.
3. Confirm that the Data Transfer Utility installed successfully.

   ```
   sudo dts --help
   ```

To install the Data Transfer Utility on Oracle Linux or Red Hat Linux

1. If you have not done so already, download the .rpm file.
2. Issue the `yum install` command as the `root` user that has write permissions to the `/opt` directory.

   ```
   sudo yum localinstall ./dts-X.Y.Z.x86_64.rpm
   ```

   `X.Y.Z` represents the version numbers that match the installer you downloaded.
3. Confirm that the Data Transfer Utility installed successfully.

   ```
   sudo dts --help
   ```

Configuring the Data Transfer Utility

Before using the Data Transfer Utility, you must create a base Oracle Cloud Infrastructure directory and two configuration files with the required credentials. One configuration file is for the data transfer administrator, the IAM user with the authorization and permissions to create

and manage transfer jobs. The other configuration file is for the data transfer upload user, the temporary IAM user that Oracle uses to upload your data on your behalf.

Base Data Transfer Directory

Create a base Oracle Cloud Infrastructure directory:

```
mkdir /root/.oci/
```

Configuration File for the Data Transfer Administrator

Create a data transfer administrator configuration file `/root/.oci/config` with the following structure:

```
[DEFAULT]
user=<The OCID for the data transfer administrator>
fingerprint=<The fingerprint of the above user's public key>
key_file=<The _absolute_ path to the above user's private key file on the host machine>
tenancy=<The OCID for the tenancy that owns the data transfer job and bucket>
region=<The region where the transfer job and bucket should exist. Valid values are:
us-ashburn-1, us-phoenix-1, eu-frankfurt-1, and uk-london-1.>
```

For example:

```
[DEFAULT]
user=ocid1.user.oc1..<unique_ID>
fingerprint=4c:1a:6f:a1:5b:9e:58:45:f7:53:43:1f:51:0f:d8:45
key_file=/home/user/ocid1.user.oc1..<unique_ID>.pem
tenancy=ocid1.tenancy.oc1..<unique_ID>
region=us-phoenix-1
```

For the data transfer administrator, you can create a single configuration file that contains different profile sections with the credentials for multiple users. Then use the `--profile` option to specify which profile to use in the command. Here is an example of a data transfer administrator configuration file with different profile sections:

```
[DEFAULT]
user=ocid1.user.oc1..<unique_ID>
fingerprint=4c:1a:6f:a1:5b:9e:58:45:f7:53:43:1f:51:0f:d8:45
key_file=/home/user/ocid1.user.oc1..<unique_ID>.pem
tenancy=ocid1.tenancy.oc1..<unique_ID>
region=us-phoenix-1
[PROFILE1]
```

```
user=ocid1.user.oc1..<unique_ID>
fingerprint=4c:1a:6f:a1:5b:9e:58:45:f7:53:43:1f:51:0f:d8:45
key_file=/home/user/ocid1.user.oc1..<unique_ID>.pem
tenancy=ocid1.tenancy.oc1..<unique_ID>
region=us-ashburn-1
```



Important

Creating an upload user configuration file with multiple profiles is *not* supported.

By default, the `DEFAULT` profile is used for all Data Transfer Utility commands. For example:

```
dts job create --compartment-id <compartment_id> --bucket <bucket_name> --display-name <display_name> --
device-type <disk_or_appliance>
```

Instead, you can issue any Data Transfer Utility command with the `--profile` option to specify a different data transfer administrator profile. For example:

```
dts job create --compartment-id <compartment_id> --bucket <bucket_name> --display-name <display_name> --
device-type <disk_or_appliance> --profile <profile_name>
```

Using the example configuration file above, the *<profile_name>* would be `profile1`.

Configuration File for the Data Transfer Upload User

Create a data transfer upload user `/root/.oci/config_upload_user` configuration file with the following structure:

```
[DEFAULT]
user=<The OCID for the data transfer upload user>
fingerprint=<The fingerprint of the above user's public key>
key_file=<The _absolute_ path to the above user's private key file on the host machine>
tenancy=<The OCID for the tenancy that owns the data transfer job and bucket>
region=<The region where the transfer job and bucket should exist. Valid values are:
us-ashburn-1, us-phoenix-1, eu-frankfurt-1, and uk-london-1.>
```

For example:

```
[DEFAULT]
user=ocid1.user.oc1..<unique_ID>
fingerprint=4c:1a:6f:a1:5b:9e:58:45:f7:53:43:1f:51:0f:d8:45
```

```
key_file=/home/user/ocid1.user.oc1..<unique_ID>.pem
tenancy=ocid1.tenancy.oc1..<unique_ID>
region=us-phoenix-1
```

Configuration File Entries

The following table lists the basic entries that are required for each configuration file and where to get the information for each entry.



Note

Data Transfer Service Does Not Support Passphrases on Private Keys

While we recommend encrypting a private key with a passphrase when generating API signing keys, Data Transfer service does not support passphrases on the key file required for the config_upload_user. If you use a passphrase, Oracle personnel cannot upload your data.

Entry	Description and Where to Get the Value	Required?
`user`	OCID of the data transfer administrator or the data transfer upload user, depending on which profile you are creating. To get the value, see Required Keys and OCIDs.	Yes
`fingerprint`	Fingerprint for the key pair being used. To get the value, see Required Keys and OCIDs.	Yes

Entry	Description and Where to Get the Value	Required?
`key_file`	Full path and filename of the private key. **Important:** The key pair must be in PEM format. For instructions on generating a key pair in PEM format, see Required Keys and OCIDs.	Yes
`tenancy`	OCID of your tenancy. To get the value, see Required Keys and OCIDs.	Yes
`region`	An Oracle Cloud Infrastructure region. See Regions and Availability Domains. Data transfer is currently supported in `us-ashburn-1`, `us-phoenix-1`, `eu-frankfurt-1`, and `uk-london-1`.	Yes

You can verify the data transfer upload user credentials using the following command:

```
dts job verify-upload-user-credentials --bucket <bucket_name>
```

Configuration File Location

The location of the configuration files is `/root/.oci/config`.

Using the Data Transfer Utility

This section provides an overview of the syntax for the Data Transfer Utility.



Important

The Data Transfer Utility must be run as the `root` user.

There are two ways you can specify Data Transfer Utility command options:

`--option <value>`

Or

`--option=<value>`

Syntax

The basic Data Transfer Utility syntax is:

```
dts <resource> <action> <options>
```

This syntax is applied, as follows:

- `dts` is the shortened utility command name
- `job` is an example of a `<resource>`
- `create` is an example of an `<action>`, and
- other utility strings are `<options>`.

The following commands to create a transfer job shows a typical Data Transfer Utility construct.

```
dts job create --compartment-id ocid1.compartment.oc1..<unique_ID> --display-name "mycompany transfer1"
--bucket mybucket --device-type appliance
```

Or:

```
dts job create --compartment-id=ocid1.compartment.oc1..<unique_ID> --display-name="mycompany transfer1"
--bucket=mybucket --device-type=appliance
```



Note

In the previous examples, provide a friendly name for the transfer job using the `--display-name` option. Avoid entering confidential information when providing resource names or descriptions.

Getting Help with Commands

You can get help using `--help`, `-h`, or `-?`. For example:

```
dts --help
```

Finding Out the Installed Version of the Data Transfer Utility

You can get the installed version of the Data Transfer Utility using `--version` or `-v`. For example:

```
dts --version
```

What's Next

You are now ready to perform data transfer-related tasks.

- For task documentation related to HDD data transfers, see Managing Disk Data Transfers.
- For task documentation related to appliance data transfers, see Managing Appliance Data Transfers.

Managing Disk Data Transfers

This topic describes in detail all the tasks required to transfer data to Oracle Cloud Infrastructure using the Data Transfer Disk service.

In preparation for data migration, an IAM user that has administrative authorization and permissions must perform the prerequisite tasks described in Preparing for Data Transfer.

The Data Transfer Utility is a command-line utility that you install on your host machine. Many data transfer tasks can be performed either using the Console or using the Data Transfer Utility. However, some data transfer tasks can only be performed using the Data Transfer Utility. This section guides you to the appropriate management interface to use for each task. See Installing and Configuring the Data Transfer Utility for detailed installation and configuration instructions.



Important

The Data Transfer Utility must be run as the `root` user.

There are two ways you can specify Data Transfer Utility parameters:

`--parameter <value>`

or

`--parameter=<value>`

Tagging Resources

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

Data Transfer currently supports applying tags to transfer jobs using the Data Transfer Utility. Tagging is not supported using the Console.

Creating a Transfer Job



Tip

You can use the Console or the Data Transfer Utility to create a transfer job.

A transfer job represents the collection of files that you want to transfer and signals the intention to upload those files to Oracle Cloud Infrastructure. A transfer job combines at least one transfer disk with a transfer package. Identify which compartment and Object Storage bucket that Oracle is to upload your data to. Create the transfer job in the same compartment as the upload bucket and supply a human-readable name for the transfer job. Avoid entering confidential information when providing transfer job names.

Creating a transfer job returns a job ID that you specify in other transfer tasks. For example:

```
ocid1.datatransferjob.region1.phx..<unique_ID>
```

To create a transfer job using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.
2. Select the designated compartment you are to use for data transfers from the drop-down list.
   A list of transfer jobs that have already been created is displayed.
3. Click **Create Transfer Job**.
4. In the **Create Transfer Job** dialog, enter a **Job Name**, and select the **Upload Bucket** from the drop-down list.
   Avoid entering confidential information in the transfer job name.
5. Select **Disk** for the **Transfer Device Type**.
6. Click **Create Transfer Job**.

To create a transfer job using the Data Transfer Utility

At the command prompt on the host, run `dts job create` to create a transfer job. The *<display_name>* is the name of the transfer job. Avoid entering confidential information in the transfer job name.

```
dts job create --bucket <bucket_name> --compartment-id <compartment_id> --display-name <display_name>
--device-type disk
```

Optionally, you can specify one or more free-form or defined tags when you create a transfer job. For more information about tagging, see Resource Tags.

To specify free-form tags when creating a job:

```
dts job create --bucket <bucket_name> --compartment-id <compartment_id> --display-name <display_name>
--device-type disk --freeform-tags '{ "<tag_key>":"<value>" }'
```

To specify defined tags when creating a job:

```
dts job create --bucket <bucket_name> --compartment-id <compartment_id> --display-name <display_name>
--device-type disk --defined-tags '{ "<tag_namespace>": { "<tag_key>":"<value>" }}'
```



Note

Tag namespaces and tag keys are created by a user with the required permissions and must already exist before you can specify them when creating a job. See Working with Defined Tags for details.

To specify multiple tags, comma separate the json-formatted additional key/value pairs:

```
dts job create --bucket <bucket_name> --compartment-id <compartment_id> --display-name <display_name>
--device-type disk --freeform-tags '{ "<tag_key>":"<value>" }', '{ "<tag_key>":"<value>" }'
```

Performing Other Transfer Job Tasks



Tip

You can use the Console or the Data Transfer Utility to perform other transfer job-related tasks.

Displaying the List of Transfer Jobs

To display the list of transfer jobs using the Console

Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.

To display the list of transfer jobs using the Data Transfer Utility

At the command prompt on the host, run `dts job list` to display the list of transfer jobs.

```
dts job list --compartment-id <compartment_id>
```

When you use the Data Transfer Utility to list jobs, tagging details are also included in the output if you specified tags.

Displaying the Details of a Transfer Job

To display the details of a transfer job using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.
2. Find the transfer job for which you want to display the details.
3. Click the Actions icon (three dots), and then click **View Details**.

To display the details of a transfer job using the Data Transfer Utility

At the command prompt on the host, run `dts job show` to display the details of a transfer job.

```
dts job show --job-id <job_id>
```

When you use the Data Transfer Utility to display the details of a job, tagging details are also included in the output if you specified tags.

Editing the Name of a Transfer Job

To edit the name of a transfer job using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.
2. Find the data transfer job that you want to edit.
3. Click the Actions icon (three dots), and then click **Edit**.
4. Edit the name of the transfer job.
   Avoid entering confidential information in the transfer job name.
5. Click **Save**.

To edit the name of a transfer job using the Data Transfer Utility

At the command prompt on the host, run `dts job update` to edit the name (`--display-name`) of a transfer job. The *<display_name>* is the new name of the transfer job. Avoid entering confidential information in the transfer job name.

```
dts job update --job-id <job_id> --display-name <display_name>
```

To edit the tags associated with a transfer job using the Data Transfer Utility

At the command prompt on the host, run `dts job update` to edit the tags associated with a transfer job. The Data Transfer Utility **replaces** any existing tags with the new key/value pairs you specify.

To edit free-form tags, provide the replacement key/value pairs:

```
dts job update --job-id <job_id> --freeform-tags '{ "<tag_key>":"<value>" }'
```

To edit defined tags, provide the replacement key value pairs:

```
dts job update --job-id <job_id> --defined-tags '{ "<tag_namespace>": { "<tag_key>":"<value>" }}'
```

To delete the tags associated with a transfer job using the Data Transfer Utility

At the command prompt on the host, run `dts job update` to delete the tags associated with a transfer job. The Data Transfer Utility **replaces** any existing tags with the new key/value pairs you specify. If you want to delete some of the tags, you would specify new tag string that does not contain the key/value pair you want to delete.

Partial tag deletion is handled in the same way as you editi tags:

- To edit free-form tags, provide the replacement key/value pairs:

```
dts job update --job-id <job_id> --freeform-tags '{ "<tag_key>":"<value>" }'
```

- To edit defined tags, provide the replacement key value pairs:

```
dts job update --job-id <job_id> --defined-tags '{ "<tag_namespace>": { "<tag_key>":"<value>" }}'
```

To delete all free-form tags:

```
dts job update --job-id <job_id> --freeform-tags '{}'
```

To delete all defined tags:

```
dts job update --job-id <job_id> --defined-tags '{}'
```

Deleting a Transfer Job

Typically, you would delete a transfer job early in the transfer process and before you create any transfer packages or disks. For example, you initiated the data transfer by creating a transfer job, but changed your mind. If you want to delete a transfer job later in the transfer process, you must first delete all transfer packages and disks associated with the transfer job.

To delete a transfer job using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.
2. Find the data transfer job that you want to delete.
3. Click the Actions icon (three dots), and then click **Delete**.
   Alternatively, you can delete a transfer job from the **View Details** page.
4. Confirm the deletion when prompted.

To delete a transfer job using the Data Transfer Utility

At the command prompt on the host, run `dts job delete` to delete a transfer job.

```
dts job delete --job-id <job_id>
```

Attaching HDDs to the Host Machine

Before creating a transfer disk from an attached HDD, remove all partitions and any file systems. To prevent the accidental deletion of data, the Data Transfer Utility doesn't work with HDDs that already have partitions or file systems. HDDs are visible to the host as block devices and must provide a valid response to the `hdparm -I <device>` Linux command.

Creating a Transfer Disk



Tip

You can only use the Data Transfer Utility to create a transfer disk.

When you create a transfer disk, the Data Transfer Utility:

- Sets up the HDD for encryption using the passphrase
- Creates a file system on the HDD
- Mounts the file system at `/mnt/orcdts_<label>`

For example:

```
/mnt/orcdts_DJZNWK3ET
```

When you register a transfer disk, Oracle Cloud Infrastructure generates a strong encryption passphrase that is used to encrypt the transfer disk. The encryption passphrase is displayed to standard output to the data transfer administrator user and cannot be retrieved again. Create a local, secure copy of the encryption passphrase, so you can reference the passphrase again.

Creating a transfer disk requires the job ID returned from when you created the transfer job and the path to the attached HDD (for example, `/dev/sdb`).

To create a transfer disk using the Data Transfer Utility

At the command prompt on the host, run `dts disk create` to create a transfer disk.

```
dts disk create --job-id <job_id> --block-device <block_device>
```

Performing Other Transfer Disk Tasks



Tip

You can only use the Data Transfer Utility to delete or cancel a transfer disk.

Deleting a Transfer Disk

Typically, you would delete a transfer disk during the disk preparation process. You created, attached, and/or copied data to the transfer disk, but have changed your mind about shipping the disk. If you want to reuse the disk, remove all file systems and create the disk again.

To delete a transfer disk using the Data Transfer Utility

At the command prompt on the host, run `dts disk delete` to delete a transfer disk.

```
dts disk delete --job-id <job_id> --disk-label <label>
```

Canceling a Transfer Disk

If you shipped a disk to Oracle, but have changed your mind about uploading the files, you can cancel the transfer disk. You can cancel a disk in a transfer package, while allowing the file upload from other disks.

Oracle cannot process canceled transfer disks. Oracle returns canceled transfer disks to the sender.

To cancel a transfer disk using the Data Transfer Utility

At the command prompt on the host, run `dts disk cancel` to cancel a transfer disk.

```
dts disk cancel --job-id <job_id> --disk-label <label>
```

Copying Data to the Transfer Disk

Attach the HDDs to the host machine and copy files to the mount point created by the Data Transfer Utility.

You can only copy regular files to transfer disks. Special files (links, sockets, pipes, and so forth) cannot be copied directly. To transfer special files, create a tar archive of the files and copy the tar archive to the transfer disk.



Note

Copy all Files Before Disconnecting a Transfer Disk

Do not disconnect a transfer disk until you copy all files from the host and generate the manifest file. If you accidentally disconnect the transfer disk before copying all files, you must unlock the disk using the encryption passphrase. The encryption passphrase was generated and displayed when you created the transfer disk. If the generated encryption passphrase is not available, you must delete the transfer disk from the transfer job and re-create the disk. All data previously copied to that transfer disk is lost.

Generating a Manifest File



Tip

You can only use the Data Transfer Utility to generate a manifest file.
The amount of time to generate the manifest file depends on the size of the upload files, disk speed, and available processing power.

After copying your data to a transfer disk, generate a manifest file using the Data Transfer Utility. The manifest contains an index of all of the copied files and generated data integrity hashes. The Data Transfer Utility copies the `config_upload_user` configuration file and referenced IAM credentials to the encrypted transfer disk. This configuration file describes the temporary IAM data transfer upload user. Oracle uses the credentials and entries defined in the `config_upload_user` file when processing the transfer disk and uploading files to Oracle Cloud Infrastructure Object Storage.



Note

Data Transfer Service Does Not Support Passphrases on Private Keys

While we recommend encrypting a private key with a passphrase when generating API signing keys, Data Transfer does not support passphrases on the key file required for the `config_upload_user`. If you use a passphrase, Oracle personnel cannot upload your data.

Oracle cannot upload data from a transfer disk without the correct credentials defined in this configuration file. See Installing and Configuring the Data Transfer Utility for more information about the required configuration files.

To create a manifest file using the Data Transfer Utility

At the command prompt on the host, run `dts disk manifest` to create a manifest file.

```
dts disk manifest --job-id <job_id> --disk-label <label>
```



Note

Do You Need to Regenerate the Manifest File?

If you add, remove, or modify any files on the disk after generating the manifest file, you must regenerate the file. If the manifest file does not match the contents of the target bucket, Oracle cannot upload the data.

Generating a Dry-Run Report of the Transfer



Tip

You can only use the Data Transfer Utility to generate a dry-run report.

You can generate a dry-run report to review the transfer results before the actual data upload. The report compares the contents of the generated manifest file with the contents of the target bucket. This report can help determine if you have duplicate files or naming collision issues.



Note

Do You Need to Regenerate the Manifest File?

If you add, remove, or modify any files on the disk after generating the manifest file, you must regenerate the file. If the manifest file does not match the contents of the target bucket, Oracle cannot upload the data.

To generate a dry-run report

At the command prompt on the host, run `dts disk dry-run` to generate a dry-run report of the transfer disk.

```
dts disk dry-run --job-id <job_id> --disk-label <label>
```

Locking a Transfer Disk



Tip

You can only use the Data Transfer Utility to lock a transfer disk.

Locking a transfer disk safely unmounts the disk and removes the encryption passphrase from the host. When you lock a transfer disk, you are prompted for the encryption passphrase that was generated when you created the transfer disk.

To lock a transfer disk using the Data Transfer Utility

At the command prompt on the host, run `dts disk lock` to lock a transfer disk.

```
dts disk lock --job-id <job_id> --disk-label <label> --block-device <block_device>
```

If you need to unlock the transfer disk, you are prompted for the encryption passphrase that was generated when you created the transfer disk.

To unlock a transfer disk using the Data Transfer Utility

At the command prompt on the host, run `dts disk unlock` to unlock a transfer disk.

```
dts disk unlock --job-id <job_id> --disk-label <label> --block-device <block_device> --encryption-passphrase <encryption_passphrase>
```

Creating a Transfer Package



Tip

You can use the Console or the Data Transfer Utility to create a transfer package.

A transfer package is the virtual representation of the physical package of disks that you are shipping to Oracle for upload to Oracle Cloud Infrastructure.

Creating a transfer package requires the job ID returned from when you created the transfer job. For example:

```
ocid1.datatransferjob.region1.phx..exampleuniqueID
```

To create a transfer package using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.
2. Find the transfer job for which you want to create a transfer package.
3. Click the Actions icon (three dots), and then click **View Details**.
   Alternatively, click the hyper-linked name of the transfer job.
   A list of transfer packages that have already been created is displayed.

4. Click **Create Transfer Package**.
5. In the **Create Transfer Package** dialog, choose the **Vendor**.
6. Click **Create Transfer Package**.

To create a transfer package using the Data Transfer Utility

At the command prompt on the host, run `dts package create` to create a transfer package.

```
dts package create --job-id <job_id>
```

Performing Other Transfer Package Tasks



Tip

You can use the Console or the Data Transfer Utility to perform other transfer package-related tasks.

Displaying the Details of a Transfer Package

To display the details of a transfer package using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.
2. Find the transfer job for which you want to see the details.
3. Click the Actions icon (three dots), and then click **View Details**.
   Alternatively, click the hyper-linked name of the transfer job.
   A list of transfer packages that have already been created is displayed.

To display the details of a transfer package using the Data Transfer Utility

At the command prompt on the host, run `dts package show` to display the details of a transfer package.

```
dts package show --job-id <job_id> --package-label <package_label>
```

Editing a Transfer Package

Edit the transfer package and supply the tracking information when you ship the package.

To edit a transfer package using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.
2. Find the transfer job for which you want to see the associated transfer packages.
3. Click the Actions icon (three dots), and then click **View Details**.
4. Find the transfer package that you want to edit.
5. Click the Actions icon (three dots), and then click **View Details**.
6. Click **Edit**.
   Change the vendor and supply the tracking information as needed.
7. Click **Edit Transfer Package**.

To edit a transfer package using the Data Transfer Utility

At the command prompt on the host, run `dts package update` to edit the details of a transfer package.

```
dts package update --job-id <job_id> --package-label <package_label> [--package-vendor <vendor_name>]
[--tracking-number <tracking_number>] [--return-tracking-number <return_tracking_number>]
```

Deleting a Transfer Package

Typically, you would delete a transfer package early in the transfer process and before you created any transfer disks. You initiated the transfer job and package, but have changed your mind. If you delete a transfer package later in the transfer process, you must first detach all associated transfer disks. You cannot delete a transfer package once the package has been shipped to Oracle.

To delete a transfer package using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.
2. Find the transfer job for which you want to see the associated transfer packages.
3. Click the Actions icon (three dots), and then click **View Details**.
4. Find the transfer package that you want to edit.
5. Click the Actions icon (three dots), and then click **View Details**.
6. Click **Edit**.
   Change the vendor and supply the tracking information as needed.
7. Click **Edit Transfer Package**.

To delete a transfer package using the Data Transfer Utility

At the command prompt on the host, run `dts package delete` to delete a transfer package.

```
dts package delete --job-id <job_id> --package-label <package_label>
```

Canceling a Transfer Package

If you shipped a transfer package, but have changed your mind about uploading the data, you can cancel a transfer package. Before canceling a transfer package, you must first cancel all transfer disks associated with that transfer package. Oracle cannot process canceled transfer packages. Oracle returns canceled transfer packages to the sender.

To cancel a transfer package using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.
2. Find the transfer job for which you want to see associated transfer packages.
3. Click the Actions icon (three dots), and then click **View Details**.
4. Find the transfer package that you want to cancel.
5. Click the Actions icon (three dots), and then click **View Details**.
6. Click **Cancel Transfer Package**.

To cancel a transfer package using the Data Transfer Utility

At the command prompt on the host, run `dts package cancel` to cancel a transfer package.

```
dts package cancel --job-id <job_id> --package-label <package_label>
```

Attaching Transfer Disks to a Transfer Package



Tip

You can use the Console or the Data Transfer Utility to attach a transfer disk to a transfer package.

You attach a transfer disk to a transfer package after you have copied your data onto the disk, generated the required manifest file, run and reviewed the dry-run report, and then locked the transfer disk in preparation for shipment.

A disk can be attached to one package, detached, and then attached to another package.

To attach a transfer disk to a transfer package using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.
2. Find the transfer job associated with the transfer package that you want to attach a disk to.
3. Click the Actions icon (three dots), and then click **View Details**.
   A list of transfer packages is displayed.
4. Find the transfer package that you want to attach a disk to.
5. Click the Actions icon (three dots), and then click **View Details**.
   Alternatively, click the hyper-linked name of the transfer package.
   A list of transfer disks is displayed.
6. Click **Attach Transfer Disks**.
7. In the **Attach Transfer Disk** dialog, select the **Transfer Disks** that you want to attach to the transfer package.
8. Click **Attach**.

To attach a transfer disk to a transfer package using the Data Transfer Utility

At the command prompt on the host, run `dts disk attach` to attach a disk to a transfer package.

```
dts disk attach --job-id <job_id> --package-label <package_label> --disk-label <label>
```

You have attached a transfer disk to a transfer package, but have changed your mind about shipping that disk with the transfer package. You can also detach a transfer disk from one transfer package and attach that disk to a different transfer package.

To detach a transfer disk to a transfer package using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and

click **Data Transfer**.

2. Find the transfer package for which you want to detach a transfer disk.
3. Click the Actions icon (three dots), and then click **View Details**.
   Alternatively, click the hyper-linked name of the transfer package.
   A list of transfer disks that have already been attached is displayed.
4. Find the transfer disk that you want to detach.
5. Click the Actions icon (three dots), and then click **View Details**.
   Alternatively, click the hyper-linked name of the transfer disk.
6. Click **Detach Transfer Disk**.

To detach a transfer disk to a transfer package using the Data Transfer Utility

At the command prompt on the host, run `dts disk attach` to attach a disk to a transfer package.

```
dts disk attach --job-id <job_id> --package-label <package_label> --disk-label <label>
```

Getting the Shipping Address for the Transfer Package



Tip

You can use the Console or the Data Transfer Utility to get the shipping address for a transfer package.

You can find the shipping address in the transfer package details.

To get the shipping address for a transfer package using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.

2. Find the transfer job for which you want to see the details.
3. Click the Actions icon (three dots), and then click **View Details**.
   Alternatively, click the hyper-linked name of the transfer job.
   A list of transfer packages that have already been created is displayed.
4. Find the transfer package for which you want to see the details.
5. Click the Actions icon (three dots), and then click **View Details**.
   Alternatively, click the hyper-linked name of the transfer job.

To get the shipping address for a transfer package using the Data Transfer Utility

At the command prompt on the host, run `dts package show` to get the shipping address for a transfer package.

```
dts package show --job-id <job_id> --package-label <package_label>
```



Important

Be sure to update the transfer package with tracking information to avoid processing delays! See Updating the Transfer Package With Tracking Information for details.

Packaging and Shipping Transfer Disks

General

Include the required return shipping label in the box when packaging transfer disks for shipment.



Note

Return Shipment Label Requirement

If you do not include the return shipping label inside the box, Oracle cannot process the transfer package.

Ensure that the transfer job and transfer package label are clearly readable on the outside of the box containing the transfer disks.



Important

If you are shipping transfer disks to `eu-frankfurt-1`, you must request that the shipping vendor requires a signature for delivery to ensure that the package is correctly delivered to Oracle.

Shipping Transfer Disks Internationally

You need to create a commercial invoice when shipping transfer disks internationally. To ensure that packages are not held up in customs, follow these guidelines when creating the commercial invoice:

- Show a unique reference number.
- Show the "bill-to party as follows:
  - For shipments to the European Union (Frankfurt) location:
    ORACLE Deutschland B.V. & Co. KG
    Riesstrasse 25
    Munich, 80992
    GERMANY
  - For shipments to the United States location:

Oracle America, Inc.
500 Oracle Parkway
REDWOOD CITY CA 94065
UNITED STATES

- Show the "ship-to party" as the address provided in the transfer package details. See Getting the Shipping Address for the Transfer Package for details.
- State that "The value shown includes the value of software and data recorded onto the hard drive unit."
- State that the "Goods are free of charge - no payment required."
- State that the type of export is "Temporary."
- Ensure that the commodity code shows the correct HS code for a hard drive unit as specified in the source country's HS code list.
- State the description as the manufacture's description of the hard drive unit and include the words "Hard Disk Drive."
- Ensure the invoice is signed and includes the printed name of the signer.

Updating the Transfer Package With Tracking Information



Tip

You can use the Console or the Data Transfer Utility to update the transfer package with tracking information.

After delivering the transfer package to the shipping vendor, update the transfer package with the tracking information.



Important

Oracle cannot process a transfer package until you update the tracking information.

To update the transfer package with tracking information using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.
2. Find the transfer job for which you want to see the associated transfer packages.
3. Click the Actions icon (three dots), and then click **View Details**.
   A list of transfer packages that have already been created is displayed.
4. Find the transfer package that you want to update.
5. Click the Actions icon (three dots), and then click **View Details**.
6. Click **Edit**.
7. Enter the **Tracking ID** and the **Return Tracking ID**.
8. Click **Edit Transfer Package**.

To update the transfer package with tracking information using the Data Transfer Utility

At the command prompt on the host, run `dts package ship` to update the transfer package tracking information.

```
dts package ship --job-id <job_id> --package-label <package_label> --package-vendor <vendor_name> --
tracking-number <tracking_number> --return-tracking-number <return_tracking_number>
```

Checking the Transfer Package Status



Tip

You can use the Console or the Data Transfer Utility to check the status of a transfer package.

When Oracle has processed the transfer disks associated with a transfer package, the status of the transfer package changes to **Processed**. When Oracle has shipped the transfer disks associated with a transfer package, the status of the transfer package changes to **Returned**.

To check the status of a transfer package using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.
2. Choose the data transfer package for which you want to display the details.
3. Click the Actions icon (three dots), and then click **View Details**.
4. Look at the **Status**.

To check the status of a transfer package using the Data Transfer Utility

At the command prompt on the host, run `dts package show` to show the status of a transfer package.

```
dts package show --job-id <job_id> --package-label <package_label>
```

Reviewing the Log Files for Each Transfer Disk

Oracle creates log files for each transfer disk uploaded. The logs are placed in the bucket where the transfer disk data was uploaded. The log file compares the transfer disk's manifest file to the contents of the target Oracle Cloud Infrastructure Object Storage bucket after file upload.

The top of the log report summarizes the overall file processing status:

```
P - Present: The file is present in both the disk and the target bucket
M - Missing: The file is present in the disk but not the target bucket. It was likely uploaded and then
deleted by another user before the summary was generated.
C - Name Collision: The file is present in the manifest but a file with the same name but different
contents is present in the target bucket.
U - Unreadable: The file is not readable from the disk
N - Name Too Long: The file name on disk is too long and could not be uploaded
```

Complete file upload details follow the summary.

```
#########################################################################################################################
#####################################  SUMMARY FOR DISK [WDH0B87L] #######################################
Generated    2017-09-08 19:46:36
TOTAL : 1110
### P present:       1110
### M missing:       0
### C name collision: 0
### U unreadable:    0
### N nameTooLong:   0
#########################################################################################################################################################################
|     STATUS     |                          NAME                          |         LAST_MODIFIED          |    SIZE(MB)    |              MD5              |                 ETag                 |
|     present    |                  small/01/T6QHX                        |  Fri Sep 08 19:04:54 UTC 2017  |      10.00     |  8clkXbwU793H2KMiaF8m6w==     |  58B2F70D1C874AAAE053824310ACAC52     |
|     present    |                  small/01/FAFUK                        |  Fri Sep 08 19:12:20 UTC 2017  |      10.00     |  8clkXbwU793H2KMiaF8m6w==     |  58B2ECE4616E46D8E053824310AC6DF6     |
|     present    |                  small/01/EVPFD                        |  Fri Sep 08 19:02:42 UTC 2017  |      10.00     |  8clkXbwU793H2KMiaF8m6w==     |  58B2ECDFA74946D6E053824310AC383A     |
|     present    |                  small/01/2UO2W                        |  Fri Sep 08 19:13:06 UTC 2017  |      10.00     |  8clkXbwU793H2KMiaF8m6w==     |
```

Closing a Transfer Job



Tip

You can use the Console or the Data Transfer Utility to close a transfer job.

Typically, you would close a transfer job when no further transfer job activity is required or possible. Closing a transfer job requires that the status of all associated transfer packages be returned, canceled, or deleted. In addition, the status of all associated transfer disks must be complete, in error, missing, canceled, or deleted.

To close a transfer job using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and

click **Data Transfer**.

2. Find the data transfer package for which you want to display the details.
3. Click the Actions icon (three dots), and then click **View Details**. Alternatively, click the hyper-linked name of the transfer job.
4. Click **Close Transfer Job**.

To close a transfer job using the Data Transfer Utility

At the command prompt on the host, run `dts job close` to close a transfer job.

```
dts job close --job-id <job_id>
```

Managing Appliance Data Transfers

This topic describes in detail all the tasks required to transfer data to Oracle Cloud Infrastructure using the Data Transfer Appliance service.

In preparation for data migration, an IAM user that has administrative authorization and permissions must perform the prerequisite tasks described in Preparing for Data Transfer.

Tagging Resources

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

Data Transfer currently supports applying tags to transfer jobs using the Data Transfer Utility. Tagging is not supported using the Console.

Preparing Your Host Machine

The Data Transfer Utility is a command-line utility that you install on a computer (host machine) at your site. Many data transfer tasks can be performed either using the Console or

using the Data Transfer Utility. However, some data transfer tasks can only be performed using the Data Transfer Utility. Each topic in this section guides you to the appropriate management interface to use for each task.

The host machine also requires terminal emulator software to access the transfer appliance serial console.

Step 1: Download and install the Data Transfer Utility on the host computer

Download and install the Data Transfer Utility on your host computer. See Installing and Configuring the Data Transfer Utility for detailed installation and configuration instructions.



Important

The Data Transfer Utility must be run as the `root` user.

Step 2: Configure the Data Transfer Utility to Use the Required Local Utilities

To export the paths for the required local utilities

At the command prompt on the host, use the `export` command to set the paths to the required utilities.

```
export OPENSSL_PATH="</path/to/binary>" # This is the full path to the openssl binary. Usually it can be found by `which openssl`

export KEYTOOL_PATH="</path/to/binary>" # This is the full path to the Java keytool binary. Usually it can be found by `which keytool`

export KEYSTORE_PATH="</path/to/binary>" # This is the full path to the Java CA certificate store (cacerts). Usually it can be found by `$(readlink /usr/bin/java | sed "s:bin/java::")lib/security/cacerts`
```

Step 3: Install and configure terminal emulator software on the host computer

Terminal emulator software provides access to the transfer appliance serial console. We recommend using the following terminal emulator software:

- PuTTY for Windows
- ZOC for OS X
- PuTTY or Minicom for Linux

Configure the terminal emulator software settings as follows:

- Baud Rate: 115200
- Emulation: VT102
- Handshaking: Disabled/off
- RTS/DTS: Disabled/off

Creating a Transfer Job



Tip

You can use the Console or the Data Transfer Utility to create a transfer job.

A transfer job represents the collection of files that you want to transfer and signals the intention to upload those files to Oracle Cloud Infrastructure. Identify which compartment and Object Storage bucket that Oracle is to upload your data to. Create the transfer job in the same compartment as the upload bucket and supply a friendly name for the transfer job. Avoid entering confidential information when providing transfer job names.

Creating a transfer job returns a job ID that you specify in other transfer tasks. For example:

```
ocid1.datatransferjob.region1.phx..<unique_ID>
```

To create a transfer job using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.

2. Select the designated compartment you are to use for data transfers from the drop-down list.
   A list of transfer jobs that have already been created is displayed.
3. Click **Create Transfer Job**.
4. In the **Create Transfer Job** dialog, enter a friendly **Job Name**, and select the **Upload Bucket** from the drop-down list.
   Avoid entering confidential information in the job name.
5. Select **Appliance** for the **Transfer Device Type**.
6. Click **Create Transfer Job**.

To create a transfer job using the Data Transfer Utility

At the command prompt on the host, run `dts job create` to create a transfer job. The *<display_name>* is the name of the transfer job. Avoid entering confidential information in the transfer job name.

```
dts job create --bucket <bucket_name> --compartment-id <compartment_id> --display-name <display_name>
--device-type appliance
```

Optionally, you can specify one or more free-form or defined tags when you create a transfer job. For more information about tagging, see Resource Tags.

To specify free-form tags when creating a job:

```
dts job create --bucket <bucket_name> --compartment-id <compartment_id> --display-name <display_name>
--device-type appliance --freeform-tags '{ "<tag_key>":"<value>" }'
```

To specify defined tags when creating a job:

```
dts job create --bucket <bucket_name> --compartment-id <compartment_id> --display-name <display_name>
--device-type appliance --defined-tags '{ "<tag_namespace>": { "<tag_key>":"<value>" }}'
```



Note

Tag namespaces and tag keys are created by a user with the required permissions and must already exist before you can specify them when creating a job. See Working with Defined Tags for details.

To specify multiple tags, comma separate the json-formatted additional key/value pairs:

```
dts job create --bucket <bucket_name> --compartment-id <compartment_id> --display-name <display_name>
--device-type appliance --freeform-tags '{ "<tag_key>":"<value>" }', '{ "<tag_key>":"<value>" }'
```

Performing Other Transfer Job Tasks



Tip

You can use the Console or the Data Transfer Utility to perform other transfer job-related tasks.

Displaying the List of Transfer Jobs

To display the list of transfer jobs using the Console

Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.

To display the list of transfer jobs using the Data Transfer Utility

At the command prompt on the host, run `dts job list` to display the list of transfer jobs.

```
dts job list --compartment-id <compartment_id>
```

When you use the Data Transfer Utility to list jobs, tagging details are also included in the output if you specified tags.

Displaying the Details of a Transfer Job

To display the details of a transfer job using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.
2. Find the transfer job for which you want to display the details.
3. Click the Actions icon (three dots), and then click **View Details**.

To display the details of a transfer job using the Data Transfer Utility

At the command prompt on the host, run `dts job show` to display the details of a transfer job.

```
dts job show --job-id <job_id>
```

When you use the Data Transfer Utility to display the details of a job, tagging details are also included in the output if you specified tags.

Editing the name of a Transfer Job

To edit the name of a transfer job using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.

2. Find the data transfer job that you want to edit.
3. Click the Actions icon (three dots), and then click **Edit**.
4. Edit the name of the transfer job.
   Avoid entering confidential information in the transfer job name.
5. Click **Save**.

To edit the name of a transfer job using the Data Transfer Utility

At the command prompt on the host, run `dts job update` to edit the name (`--display-name`) of a transfer job. The *<display_name>* is the new name of the transfer job. Avoid entering confidential information in the transfer job name.

```
dts job update --job-id <job_id> --display-name <display_name>
```

To edit the tags associated with a transfer job using the Data Transfer Utility

At the command prompt on the host, run `dts job update` to edit the tags associated with a transfer job. The Data Transfer Utility **replaces** any existing tags with the new key/value pairs you specify.

To edit free-form tags, provide the replacement key/value pairs:

```
dts job update --job-id <job_id> --freeform-tags '{ "<tag_key>":"<value>" }'
```

To edit defined tags, provide the replacement key value pairs:

```
dts job update --job-id <job_id> --defined-tags '{ "<tag_namespace>": { "<tag_key>":"<value>" }}'
```

To delete the tags associated with a transfer job using the Data Transfer Utility

At the command prompt on the host, run `dts job update` to delete the tags associated with a transfer job. The Data Transfer Utility **replaces** any existing tags with the new key/value pairs you specify. If you want to delete some of the tags, you would specify new tag string that does not contain the key/value pair you want to delete.

Partial tag deletion is handled in the same way as you editi tags:

- To edit free-form tags, provide the replacement key/value pairs:

```
dts job update --job-id <job_id> --freeform-tags '{ "<tag_key>":"<value>" }'
```

- To edit defined tags, provide the replacement key value pairs:

```
dts job update --job-id <job_id> --defined-tags '{ "<tag_namespace>": { "<tag_key>":"<value>" }}'
```

To delete all free-form tags:

```
dts job update --job-id <job_id> --freeform-tags '{}'
```

To delete all defined tags:

```
dts job update --job-id <job_id> --defined-tags '{}'
```

Deleting a Transfer Job

You can only delete a transfer job early in the transfer process. For example, you initiated the data transfer by creating a transfer job, but changed your mind. If you want to delete a transfer job after you requested transfer appliances, you must first delete those appliance requests.

To delete a transfer job using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.
2. Find the data transfer job that you want to delete.

3. Click the Actions icon (three dots), and then click **Delete**.
   Alternatively, you can delete a transfer job from the **View Details** page.
4. Confirm the deletion when prompted.

To delete a transfer job using the Data Transfer Utility

At the command prompt on the host, run `dts job delete` to delete a transfer job.

```
dts job delete --job-id <job_id>
```

Requesting a Transfer Appliance



Tip

You can use the Console or the Data Transfer Utility to request a transfer appliance.

Oracle Cloud Infrastructure customers can use data transfer appliances to migrate data for free. You are only charged for Object Storage usage once the data is successfully transferred to your designated bucket. All appliance requests still require approval from Oracle.



Tip

We recommend that you identify the data you intend to upload and make data copy preparations before requesting the transfer appliance.

Creating a transfer appliance request returns an Oracle-assigned appliance label. For example:

```
XA8XM27EVH
```

To request a transfer appliance using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.
   Choose the transfer job that you want to request a transfer appliance for.
2. Under **Transfer Appliances**, click **Request Transfer Appliance**.
3. In the **Request Transfer Appliance** dialog box, provide the shipping address details where you want the appliance sent.
   - **Company Name:** Required. Specify the name of the company that owns the data being migrated to Oracle Cloud Infrastructure.
   - **Recipient Name:** Required. Specify the name of the recipient to send the appliance to.
   - **Recipient Phone Number:** Required. Specify the recipient's phone number.
   - **Recipient Email Address:** Required. Specify the recipient's email address.
   - **Care Of:** Optional intermediary party responsible for transferring the appliance shipment from the delivery vendor to the intended recipient.
   - **Address Line 1:** Required. Specify the street address to send the appliance to.
   - **Address Line 2:** Optional identifying address details like building, suite, unit, or floor information.
   - **City/Locality:** Required. Specify the city or locality.
   - **State/Province/Region:** Required. Specify the state, province, or region.
   - **Zip/Postal Code:** Specify the zip code or postal code.
   - **Country:** Required. Select the country.
4. Click **Request Transfer Appliance**.

To request a transfer appliance using the Data Transfer Utility

At the command prompt on the host, run `dts appliance request` to request a data transfer appliance.

Here are the minimum requirements for the transfer appliance request:

```
dts appliance request --job-id <job_id> --addressee <addressee> --address1 <address_line1> --city-or-
locality <city_or_locality> --state-or-region <state_or_region> --country <country> --zip-code <zip>
```

In addition, you can specify these optional fields in the request:

```
--care-of <care_of>
--address2 <address_line2>, --address3 <address_line3>, and --address4 <address_line4>
--email <email_address>
--phone-number <phone_number>
--profile <profile>
```

When you submit an appliance request, Oracle generates a unique label (name) to identify the transfer appliance and your request is sent to Oracle for approval and processing.

Monitoring the Status of Your Request for a Transfer Appliance

The time it takes to approve, prepare, and ship your transfer appliance request varies and depends on various factors, including current available inventory. Oracle provides status updates daily throughout the transfer appliance request and ship process.

To monitor the status of your transfer appliance request using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.
2. Find and select the transfer job for which you want to monitor associated appliance requests.
3. Under **Transfer Appliances**, find the appliance label Oracle assigned to your data transfer appliance request and look at the **Status** field.

Here are the key status values to look for when monitoring your transfer appliance request:

- **Requested**: You successfully completed your request for a transfer appliance. The status displays **Requested** until Oracle approves your transfer appliance request.
- **Rejected**: Oracle denied your transfer appliance request.

> **Important**
>
> If your appliance request is denied and you have questions, contact your Sales Representative or file a Service Request (SR).

- **Oracle Preparing**: Oracle approved your transfer appliance request. The status displays **Oracle Preparing** until the transfer appliance is shipped to you.
- **Shipping**: Oracle completed the necessary preparations and shipped your transfer appliance. When the appliance is shipped, Oracle provides the serial number of the appliance, the shipping vendor, and the tracking number in the **Transfer Appliance Details**. The status displays **Shipping** until the appliance is delivered to you.
- **Delivered**: The shipping vendor delivered your transfer appliance. When the appliance is delivered, Oracle provides the date and time the appliance was received in the **Transfer Appliance Details**. The status displays **Delivered**.

To monitor the status of your transfer appliance request using the Data Transfer Utility

At the command prompt on the host, run `dts appliance show` to monitor your transfer appliance request.

```
dts appliance show --job-id <job_id> --appliance-label <label>
```

See Data Transfer Appliance Statuses for an explanation of all transfer appliance statuses.

Performing Other Transfer Appliance Tasks



Tip

You can use the Console or the Data Transfer Utility to perform other transfer appliance-related tasks.

Displaying the Details of a Transfer Appliance

To display the details of a transfer appliance using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.
2. Find the transfer job for which you want to display the details of an associated transfer appliance.
   The list of transfer appliances is displayed below the transfer job details.
3. Find the transfer appliance for which you want to display the details.
4. Click the Actions icon (three dots), and then click **View Details**.

To display the details of a transfer appliance using the Data Transfer Utility

At the command prompt on the host, run `dts appliance show` to display the details of a transfer appliance.

```
dts appliance show --job-id <job_id> --appliance-label <label>
```

Editing the Appliance Request Shipping Information

You can only edit the shipping information when the status is **Requested**.

To edit the appliance request shipping information using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.
2. Find the **Requested** transfer appliance that you want to edit the shipping information.
3. Click the Actions icon (three dots), and then click **Edit**.
4. Edit the shipping information for the transfer appliance.
5. Click **Save**.

To edit the appliance request shipping information using the Data Transfer Utility

At the command prompt on the host, run `dts appliance update` to edit the shipping information for the transfer appliance.

```
dts appliance update-shipping-address --job-id <job_id> --appliance-label <label> <changed_fields>
```

The *<changed_fields>* variable represents one or more of the following shipping address fields that you want to update:

```
--addressee <addressee> --careOf <care_of> --address1 <street_address> --city <city> --state
<addressee> --zip <zip> --country <country> --phone <phone>
```

Deleting a Transfer Appliance Request

You can delete a transfer appliance request before Oracle approves the request—the status must be **Requested**. For example, you initiated the transfer by creating a transfer job and requested a transfer appliance, but changed your mind.

To delete a transfer appliance request using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.
2. Find the data transfer job and transfer appliance request that you want to delete.

3. Click the Actions icon (three dots), and then click **Delete**.
   Alternatively, you can delete a transfer appliance request from the **Transfer Appliance Details** page.
4. Confirm the deletion when prompted.

To delete a transfer appliance request using the Data Transfer Utility

At the command prompt on the host, run `dts appliance delete` to delete a transfer appliance.

```
dts appliance delete --job-id <job_id> --appliance-label <label>
```

Unpacking and Preparing Your Transfer Appliance

When the shipping vendor delivers your transfer appliance, Oracle updates the status as **Delivered** and provides the date and time the appliance was received in the **Transfer Appliance Details**.



Important

Your transfer appliance arrives in a transit case with a telescoping handle and wheels. The case amenities allow for easy movement to the location where you intend to place the appliance to upload your data.

Retain all packaging materials! When shipping the transfer appliance back to Oracle, you must package the appliance in the same manner and packaging in which the appliance was received.

Here are the tasks involved in unpacking and getting your transfer appliance ready to configure.

1. Inspect the tamper-evident security tie on the transit case.
   If the appliance was tampered with during transit, the tamper-evident security tie serves to alert you.

   Warning

   If the security tie is damaged or is missing, do not plug the appliance into your network! Immediately file a Service Request (SR).

2. Remove and compare the number on the security tie with the number logged by Oracle.

To see the security tie number logged by Oracle using the Console

a. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.
b. Find the transfer job and transfer appliance associated with the removed security tie.
c. Click the **Actions** icon (•••), and then click **View Details**.
d. Look at the contents of the **Send Security Tie ID** field in the **Transfer Appliance Details** and compare that number with the number on the physical tag.

To see the security tie number logged by Oracle using the Data Transfer Utility

At the command prompt on the host, run `dts appliance show` to delete a transfer appliance.

```
dts appliance show --job-id <job_id> --appliance-label <label>
```



Warning

If the number on the physical security tie does not match the number logged by Oracle, do not plug the appliance into your network! Immediately file a Service Request (SR).

3. Open the transit case and ensure that the case contains the following items:
    - Appliance unit and power cable (two types of power cables provided: C14 and C13 to 14)
    - USB to DB-9 serial cable
    - Return shipping instructions (retain these instructions)
    - Return shipping label, label sleeve, tie-on tag, and zip tie
    - Return shipment tamper-evident security tie (use this tie to ensure secure transit case back to Oracle)
4. Compare the number on the return shipment security tie with the number logged by Oracle.

To see the security tie number logged by Oracle using the Console

a. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.

b. Find the transfer job and transfer appliance associated with the return shipment security tie.

c. Click the Actions icon (three dots), and then click **View Details**.

d. Look at the contents of the **Return Security Tie ID** field in the **Transfer Appliance Details** and compare that number with the number on the physical tag.

To see the security tie number logged by Oracle using the Data Transfer Utility

At the command prompt on the host, run `dts appliance show` to the security tie number associated with the transfer appliance.

```
dts appliance show --job-id <job_id> --appliance-label <label>
```



Warning

If the number on the return security tie does not match the number logged by Oracle, file a Service Request (SR). These security tie numbers must match or Oracle cannot upload data from your returned transfer appliance.

5. Remove the transfer appliance from the case and place the appliance on a solid surface or in a rack.



Warning

We recommend assistance lifting the transfer appliance out of the transit case and placing the appliance in a rack or on a desk top. The total shipping weight is about 64 lbs (29.0299 kg) and appliance weight is 38 lbs (17.2365 kg).

6. Connect the appliance to your local network using one of the following:
   - 10GBase-T: Standard RJ-45
   - SFP+: The transceiver must be compatible with Intel X520 NICs.

7. Attach one of the provided power cords to the appliance and plug the other end into a grounded power source.
8. Turn on the appliance by flipping the power switch on the back of the appliance.
9. Connect the appliance to your host computer using the provided USB to DB-9 serial cable.



Note

You might need to download the driver for this cable on your host computer: https://www.cablestogo.com/product/26887/5ft-usb-to-db9-male-serial-rs232-adapter-cable#support

Configuring Your Transfer Appliance Networking

When the appliance boots up, an appliance serial console configuration menu is displayed on the terminal emulator on the host machine. It can take up to 5 minutes for the serial console menu to display.

The transfer appliance supports a single active network interface on any of the 10-Gbps network ports. If only one interface is cabled and active, that interface is chosen automatically. If multiple interfaces are active, you are given the choice to select the interface to use.

To configure your transfer appliance networking

1. Using the terminal emulator on the host machine, select **Configure Networking** from the appliance serial console menu.

2. Provide the required networking information when prompted:
   - **IP Address:** IP address of the transfer appliance.
   - **Subnet Mask Length:** The count of leading 1 bits in the subnet mask. For example, if the subnet mask is 255.255.255.0 then the length is 24.
   - **Default Gateway:** Default gateway for network communications.

   For example:

```
Configure Networking:
^C to cancel

Configuring IP address, subnet mask length, gateway
Example:
IP Address : 10.0.0.2
Subnet Mask Length : 24
Gateway : 10.0.0.1
Subnet mask length 24 <=> 255.255.255.0IP

Address: 10.0.0.1
Subnet Mask Length: 24
Gateway: 10.0.0.1

Configuring IP address 10.0.0.1 netmask 255.255.255.0 default gateway 10.0.0.1
Enabling enp0s3
Now trying to restart the network

Network configuration is complete

New authentication material created.

Client access token                  : 4iH1gw1okPJO
Appliance certificate MD5 fingerprint : BF:C6:49:9B:25:FE:9F:64:06:7E:DF:F5:F9:E5:C6:56
Press ENTER to return...
```

When you configure a network interface, the appliance software generates a new client access token and appliance X.509/SSL certificate. The access token is used to authorize your host machine to communicate with the Data Transfer Appliance's Management Service. The x.509/SSL certificate is used to encrypt communications with the Data Transfer Appliance's Management Service over the network. You will need provide the access token and SSL

certificate fingerprint values displayed here when you use the Data Transfer Utility to initialize authentication on your host machine.

You can change the selected interface, network information, and reset the authentication material at any time by selecting **Configure Networking** again from the appliance serial console menu.

Setting Up Your Host Machine

This topic describes the tasks required to set up your host machine to communicate with the transfer appliance using the Data Transfer Appliance Management Service. This Management Service controls all functional aspects of the transfer appliance.



Tip

You can only use the Data Transfer Utility to set up your host machine.

Step 1: Initializing Authentication

Initialize authentication to allow the host machine to communicate with the data transfer appliance. Use the values returned from the **Configure Networking** command. See Configuring Your Transfer Appliance Networking for details.

To initialize authentication

At the command prompt on the host, run `dts physical-appliance initialize-authentication` to initialize authentication.

```
dts physical-appliance initialize-authentication --appliance-cert-fingerprint="<fingerprint>" --
appliance-ip="<ip_address>"
```

For example:

```
dts physical-appliance initialize-authentication --appliance-cert-
fingerprint="F7:1B:D0:45:DA:04:0C:07:1E:B2:23:82:E1:CA:1A:E9" --appliance-ip="10.0.0.1"
```

When prompted, supply the access token and system Java keystore password. The appliance access token is stored in the Data Transfer Utility configuration and the x.509/SSL certificate is stored in the system Java keystore. Ask your administrator or vendor what the password is for your system Java keystore. For example:

```
Access token ('q' to quit): p5jZuonkmYsz1mhI
The X.509/SSL certificate for the appliance must be added to the local Java Keystore.
Continue? [y/n] y
Java Keystore password ('q' to quit): <keystore_password>
Running Java keytool to delete any appliance certificate from the keystore ...
Running Java keytool to add appliance certificate to the keystore ...
Appliance certificate added.
```

The host machine can now communicate with the transfer appliance.

To show the status of and storage details about the connected appliance

At the command prompt on the host, run `dts physical-appliance show` to show the status of the connected appliance.

```
dts physical-appliance show
```

For example:

```
Appliance Info :
 encryptionConfigured : true
 lockStatus : NOT_LOCKED
 finalizeStatus : NOT_FINALIZED
 totalSpace : 64.56 GB
 availableSpace : 64.53 GB
```

Step 2: Configuring Appliance Encryption

You must configure the appliance to use encryption. Oracle Cloud Infrastructure creates a strong passphrase for each appliance. The Data Transfer Utility securely collects the strong passphrase from Oracle Cloud Infrastructure and sends that passphrase to the Data Transfer service.

If your environment requires Internet-aware applications to use network proxies, ensure that you set up the required Linux environment variables. See the HTTP proxy information in the Data Transfer Utility topic.



Important

If you are working with multiple appliances at the same time, be sure the job ID and appliance label you specify in this step matches the physical appliance you are currently working with. You can get the serial number associated with the job ID and appliance label using the Data Transfer Utility or the Console. You can find the serial number of the physical appliance on the back of the device on the agency label.

To configure transfer appliance encryption

At the command prompt on the host, run `dts physical-appliance configure-encryption` to configure appliance encryption.

```
dts physical-appliance configure-encryption --job-id <job_id> --appliance-label <label>
```

For example:

```
dts physical-appliance configure-encryption --job-id
"ocid1.datatransferjob.region1.phx..exampleuniqueID" --appliance-label "XA8XM27EVH"
```

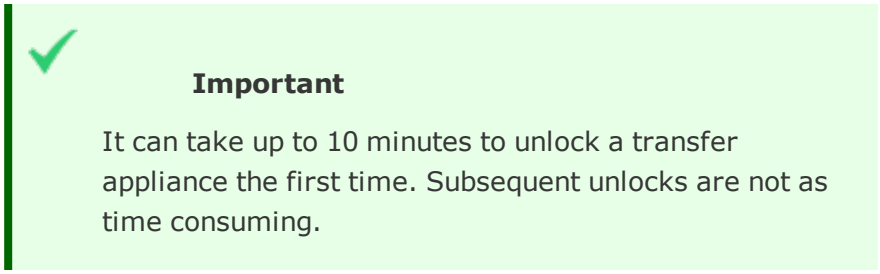
Step 3: Unlocking the Transfer Appliance

Before you can write data to the transfer appliance, you must unlock the appliance. Unlocking the transfer appliance requires the strong passphrase that is created by Oracle Cloud Infrastructure for each appliance. Unlocking can be accomplished in two different ways:

- If you provide the `--job-id` and `--appliance-label` when running the `unlock` command, the Data Transfer Utility retrieves the passphrase from Oracle Cloud

Infrastructure and sends it to the transfer appliance during the unlock operation.

- You can query Oracle Cloud Infrastructure for the passphrase and provide that passphrase when prompted during the unlock operation.



Important

It can take up to 10 minutes to unlock a transfer appliance the first time. Subsequent unlocks are not as time consuming.

To have the Data Transfer Utility retrieve the passphrase to unlock the transfer appliance

At the command prompt on the host, run `dts physical-appliance unlock` with `--job-id` and `--appliance-label` to unlock the appliance.

```
dts physical-appliance unlock --job-id <job_id> --appliance-label <label>
```

For example:

```
dts physical-appliance unlock --job-id
"ocid1.datatransferjob.region1.phx.exampleccsoazstiksiyurk7f5xhveazrbzdpwhqiwwuklxw3g5hb4movoa" --
appliance-label "XA8XM27EVH"
```

To query Oracle Cloud Infrastructure for the passphrase to provide to unlock the transfer appliance

At the command prompt on the host, run `dts appliance get-passphrase` to obtain the passphrase from the Oracle Cloud Infrastructure.

```
dts appliance get-passphrase --job-id <job_id> --appliance-label <label>
```

Then, run `dts physical-appliance unlock` without `--job-id` and `--appliance-label` and supply the passphrase when prompted.

```
dts physical-appliance unlock
```

Writing Data to the Transfer Appliance

Appliance data transfer supports NFS-v3 to write data to the appliance. In preparation for writing data, create and configure a dataset to write to. A dataset is a collection of files that are treated similarly. You can write up to 100 million files onto the appliance for transfer. We currently support one dataset per transfer appliance.



Important

Files written to the appliance must be world readable and the directories must be world readable and world executable. If files and directories do not match this criteria, you will not be able to seal the appliance.

You can only copy regular files to transfer appliances. Special files (for example, links, sockets, and pipes) cannot be copied directly. To transfer special files, create a tar archive of these files and copy the tar archive to the transfer appliance.



Tip

You can only use the Data Transfer Utility to create, configure, and allow access to a dataset over NFS.

Step 1: Creating an NFS dataset

To create an NFS dataset

At the command prompt on the host, run `dts nfs-dataset create` to create an NFS dataset.

```
dts nfs-dataset create --name <dataset_name>
```

For example:

```
dts nfs-dataset create --name nfs-ds-1
```

Step 2: Configuring export settings on the dataset

To configure export settings on an NFS dataset

At the command prompt on the host, run `dts nfs-dataset set-export` to configure export settings on an NFS dataset.

```
dts nfs-dataset set-export --name <dataset_name> --rw=true --world=true
```

For example:

```
dts nfs-dataset set-export --name nfs-ds-1 --rw=true --world=true
```

Here is another example of creating the export to give read/write access to a subnet:

```
dts nfs-dataset set-export --name nfs-ds-1 --ip=10.0.0.0 --subnet-mask-length=24 --rw=true --world=false
```

Step 3: Activating the dataset

Activation creates the NFS export, making the dataset accessible to NFS clients.

To activate the NFS dataset

At the command prompt on the host, run `dts nfs-dataset activate` to activate the NFS dataset.

```
dts nfs-dataset activate --name <dataset_name>
```

For example:

```
dts nfs-dataset activate --name nfs-ds-1
```

Step 4: Confirming that the NFS share is visible

To confirm that the NFS share is visible

At the command prompt on the host, use the `showmount` command to confirm that the NFS share is visible.

```
showmount -e <appliance_ip>
```

For example:

```
showmount -e 10.0.0.1
Export list for 10.0.0.1:
/data/nfs-ds-1 *
```

Step 5: Mounting the NFS share

To mount the NFS share

At the command prompt on the host, use the `mount` command to mount the NFS share.

```
mount -t nfs <appliance_ip>:/data/<dataset_name> <mountpoint>
```

For example:

```
mount -t nfs 10.0.0.1:/data/nfs-ds-1 /mnt/nfs-ds-1
```

After the NFS share is mounted, you can write data to the share.

Step 6: Writing data to the share

Copy your data to the transfer appliance using normal file system tools.



Important

Files written to the appliance must be world readable and the directories must be world readable and world executable. If files and directories do not match this criteria, you will not be able to seal the appliance.

Step 7: Deactivating the dataset

After you are done writing data, deactivate the dataset. Deactivation removes the NFS export on the dataset, disallowing any further writes.

To deactivate the NFS dataset

At the command prompt on the host, run `dts nfs-dataset deactivate` to deactivate the NFS dataset.

```
dts nfs-dataset deactivate --name <dataset_name>
```

For example:

```
dts nfs-dataset deactivate --name nfs-ds-1
```

Step 8: Sealing the dataset

Sealing a dataset stops all writes to the dataset. Sealing a dataset is a long running process that can take some time to complete. The completion time depends upon the number of files and total amount of data that was copied to the transfer appliance.

If you issue the `seal` command without the `--wait` option, the seal operation is triggered and runs in the background. You are returned to the command prompt and can use the `seal-status` command to monitor the sealing status. If you issue the `seal` command with the `--wait` option, the seal operation is triggered and continues to provide status updates until sealing completion.



Important

Files written to the appliance must be world readable and the directories must be world readable and world executable. If files and directories do not match this criteria, you will not be able to seal the appliance.

The sealing operation generates a manifest across all files in the dataset. The manifest contains an index of all of the copied files and generated data integrity hashes.

If the seal process fails with reasons like `Failed to walk filesystem - java.nio.file.AccessDeniedException: /data/la-rc-ds/dir1/dir2` or `Number of files with no read perms : 1`, reactivate the dataset and correct permissions for the files and directories.

To seal the NFS dataset

At the command prompt on the host, run `dts nfs-dataset seal` to seal the NFS dataset.

```
dts nfs-dataset seal --name <dataset_name> [--wait]
```

For example:

```
dts nfs-dataset seal --name nfs-ds-1
Seal initiated. Please use seal-status command to get progress.
```

To monitor the dataset sealing process

At the command prompt on the host, run `dts nfs-dataset seal-status` to monitor the dataset sealing process.

```
dts nfs-dataset seal-status --name <dataset_name>
```

For example, here is the status that is issued upon sealing completion:

```
dts nfs-dataset seal-status --name nfs-ds-1

Seal Status :
```

```
success           : true
failureReason     : *** none ***
startTime         : 2018/07/10 18:24:05 EDT
endTime           : 2018/07/10 18:24:06 EDT
numFilesToProcess : 2000
numFilesProcessed : 2000
bytesToProcess    : 1.95 GB
bytesProcessed    : 1.95 GB
```

Finalizing the Transfer Appliance



Tip

You can only use the Data Transfer Utility to finalize the transfer appliance.

Finalizing an appliance tests and copies the following to the transfer appliance:

- Upload user configuration credentials
- Private PEM key details
- Name of the upload bucket

The credentials, certificate, and bucket are required for Oracle to be able to upload your data to Oracle Cloud Infrastructure Object Storage. When you finalize a transfer appliance, you can no longer access the appliance for dataset operations unless you unlock the appliance. See Reopening a Dataset if you need to unlock an appliance that was finalized.



Important

If you are working with multiple appliances at the same time, be sure the job ID and appliance label you specify in this step matches the physical appliance you are currently working with. You can get the serial number associated with the job ID and appliance label using the Data Transfer Utility or the Console. You can find the serial number of the physical appliance on the back of the device on the agency label.

To finalize the appliance

1. Deactivate and seal the dataset before finalizing the appliance.
2. At the command prompt on the host, run `dts physical-appliance finalize` to finalize a transfer appliance.

```
dts physical-appliance finalize --job-id <job_id> --appliance-label <label>
```

For example:

```
dts physical-appliance finalize --job-id "ocid1.datatransferjob.region1.phx..exampleuniqueID" --
appliance-label "XA8XM27EVH"
```

Reopening a Dataset



Tip

You can only use the Data Transfer Utility to reopen a dataset.

If changes are necessary after sealing a dataset or finalizing an appliance, you must reopen the dataset to modify the contents. Make the required changes and again seal the dataset. Resealing the dataset generates a new manifest.



Note

If an appliance is rebooted or power cycled, follow the instructions in this topic to reopen the dataset.

Step 1: Unlocking the Transfer Appliance

Before you can write data to the transfer appliance, you must unlock the appliance. Unlocking the transfer appliance requires the strong passphrase that is created by Oracle Cloud Infrastructure for each appliance. Unlocking can be accomplished in two different ways:

- If you provide the `--job-id` and `--appliance-label` when running the `unlock` command, the Data Transfer Utility retrieves the passphrase from Oracle Cloud Infrastructure and sends it to the transfer appliance during the unlock operation.
- You can query Oracle Cloud Infrastructure for the passphrase and provide that passphrase when prompted during the unlock operation.

To have the Data Transfer Utility retrieve the passphrase to unlock the transfer appliance

At the command prompt on the host, run `dts physical-appliance unlock` with `--job-id` and `--appliance-label` to unlock the appliance.

```
dts physical-appliance unlock --job-id <job_id> --appliance-label <label>
```

For example:

```
dts physical-appliance unlock --job-id
"ocid1.datatransferjob.region1.phx.exampleccsoazstiksiyurk7f5xhveazrbzdpwhqiwwuklxw3g5hb4movoa" --
appliance-label "XA8XM27EVH"
```

To query Oracle Cloud Infrastructure for the passphrase to provide to unlock the transfer appliance

At the command prompt on the host, run `dts appliance get-passphrase` to obtain the passphrase from the Oracle Cloud Infrastructure.

```
dts appliance get-passphrase --job-id <job_id> --appliance-label <label>
```

Then, run `dts physical-appliance unlock` without `--job-id` and `--appliance-label` and supply the passphrase when prompted.

```
dts physical-appliance unlock
```

Step 2: Reopening the Transfer Appliance

Reopen the dataset to write data to the transfer appliance again.

To reopen an NFS dataset

At the command prompt on the host, run `dts nfs-dataset reopen` to reopen an NFS dataset.

```
dts nfs-dataset reopen --name <dataset_name>
```

Step 3: Repeat Steps to Write Data to the Appliance

Repeat the same tasks you performed when you originally wrote data to the appliance beginning with activating the dataset in the Writing Data to the Transfer Appliance section.

Shutting Down the Transfer Appliance

Shut down the appliance before packing up and shipping the appliance back to Oracle.

To shut down the appliance

Using the terminal emulator on the host machine, select **Shutdown** from the appliance serial console.

Packing and Shipping Transfer Appliance to Oracle

Return the appliance to Oracle within 30 days. If you need the transfer appliance beyond the standard 30-day window, you can file a Service Request (SR) to ask for an extension of up to 60 days.



Important

Review and follow the instructions that were provided in the transit case with the appliance.

To pack and ship the appliance

1. Unplug the power cord from the power source and detach the other end of the cord from the appliance.
2. Disconnect the appliance from your network.
3. Remove the return shipment tamper-evident security tie from the transit case.
4. Place the transfer appliance, power cord, and serial cable in the transit case.



Warning

We recommend assistance lifting and placing the transfer appliance back into the transit case. The total shipping weight is about 64 lbs (29.0299 kg) and appliance weight is 38 lbs (17.2365 kg).

5. Close and secure the transit case with the return tamper-evident security tie.
6. Loop the top of the plastic tie-on tag with return shipping label through the handle of the transit case. Remove the protective tape from the back of the tie-on tag, exposing the adhesive area to secure the tag onto itself. Use the provided zip tie to secure the tie-on

tag to the handle.

7. Return the transit case to FedEx by doing one of the following:
   - Drop off the packed, sealed, and labeled transit case to an FedEx Authorized ShipCenter location or a nearby FedEx Office location. *Obtain a receipt from the vendor to certify transfer of custody.*
   - Schedule a pickup with FedEx at your location. Ensure that the transit case is packed, sealed, and labeled before FedEx arrives for pickup.

   The shipping vendor notifies Oracle when the transfer appliance is shipped back to Oracle for upload to Oracle Cloud Infrastructure Object Storage.

Canceling a Transfer Appliance

If you change your mind about uploading your data to Oracle Cloud Infrastructure, you must cancel the transfer appliance. You can only cancel a transfer appliance after you ship the appliance back to Oracle. You can cancel one transfer appliance, while allowing the file upload from other appliances.

Oracle does not process canceled transfer appliances. Oracle retains the canceled transfer appliance and wipes all the data from the device.

To cancel a transfer device using the Data Transfer Utility

At the command prompt on the host, run `dts appliance cancel` to cancel a transfer device.

```
dts appliance cancel --job-id <job_id> --appliance-label <label>
```

Monitoring the Status of Your Transfer Appliance Return Shipment

The shipping vendor notifies Oracle when your transfer appliance is picked up and shipped back for upload to Oracle Cloud Infrastructure Object Storage.

To monitor the status of your transfer appliance return shipment using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.
2. Find the transfer job and transfer appliance that you shipped back to Oracle for data upload.
3. Under **Transfer Appliances**, look at the **Status** field.

Here are the key status values to look for when monitoring your transfer appliance return shipment:

- **Return Shipped**: You shipped your transfer appliance back to Oracle. The status displays **Return Shipped** until Oracle receives your transfer appliance.
- **Oracle Received**: Oracle received your transfer appliance shipment. The status displays **Oracle Received** until Oracle processes and uploads the data from your transfer appliance.
- **Oracle Received Canceled**: You canceled your transfer appliance after you shipped the appliance back to Oracle. Oracle received your canceled transfer appliance. Oracle does *not* upload the appliance data.
- **Processing**: Oracle is processing and uploading data from your transfer appliance. The status displays **Processing** until the transfer appliance upload is complete.
- **Complete**: Oracle completed your transfer appliance data upload. Your data is available in your designated bucket in Oracle Cloud Infrastructure Object Storage.

Reviewing the Transfer Appliance Log Files

Oracle creates transfer log files for each transfer appliance uploaded. The logs are placed in the bucket where the transfer device data was uploaded. The log file compares the transfer appliance's manifest file to the contents of the target Oracle Cloud Infrastructure Object Storage bucket after file upload.

> **Note**
>
> If you chose to upload your data to an Archive Storage bucket, you must first restore the log file object before you can download that file for review.

The top of the log report summarizes the overall file processing status:

```
P - Present: The file is present in both the device and the target bucket
M - Missing: The file is present in the device but not the target bucket. It was likely uploaded and
then deleted by another user before the summary was generated.
C - Name Collision: The file is present in the manifest but a file with the same name but different
contents is present in the target bucket.
U - Unreadable: The file is not readable from the disk
N - Name Too Long: The file name on disk is too long and could not be uploaded
```

Complete file upload details follow the summary.

```
##########################################################################################################################################
########################################  SUMMARY FOR DISK [WDH0B87L] ####################################################
Generated    2017-09-08 19:46:36
TOTAL : 1110
### P present:       1110
### M missing:       0
### C name collision: 0
### U unreadable:    0
### N nameTooLong:   0
##########################################################################################################################################################################################
|    STATUS     |                        NAME                        |        LAST_MODIFIED          |    SIZE(MB)    |            MD5             |                ETag                |
|    present    |                   small/01/T6QHX                   |  Fri Sep 08 19:04:54 UTC 2017 |     10.00      |  8clkXbwU793H2KMiaF8m6w==  |  58B2F70D1C874AAAE053824310ACAC52   |
|    present    |                   small/01/FAFUK                   |  Fri Sep 08 19:12:20 UTC 2017 |     10.00      |  8clkXbwU793H2KMiaF8m6w==  |  58B2ECE4616E46D8E053824310AC6DF6   |
|    present    |                   small/01/EVPFD                   |  Fri Sep 08 19:02:42 UTC 2017 |     10.00      |  8clkXbwU793H2KMiaF8m6w==  |  58B2ECDFA74946D6E053824310AC383A   |
|    present    |                   small/01/2UO2W                   |  Fri Sep 08 19:13:06 UTC 2017 |     10.00      |  8clkXbwU793H2KMiaF8m6w==  |
```

If you upload more than 100,000 files, the upload details are broken into multiple pages. You can only download the first page from the Console. Download the rest of the pages directly from the Object Storage bucket. The subsequent pages have the same object name as the first page, but have an enumerated suffix.

Closing a Transfer Job



Tip

You can use the Console or the Data Transfer Utility to close a transfer job.

Typically, you would close a transfer job when no further transfer job activity is required or possible. Closing a transfer job requires that the status of all associated transfer appliances be returned, canceled, or deleted.

To close a transfer job using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.
2. Find the data transfer package for which you want to display the details.
3. Click the Actions icon (three dots), and then click **View Details**.
   Alternatively, click the hyper-linked name of the transfer job.
4. Click **Close Transfer Job**.

To close a transfer job using the Data Transfer Utility

At the command prompt on the host, run `dts job close` to close a transfer job.

```
dts job close --job-id <job_id>
```

Data Transfer Appliance Statuses

Oracle provides status updates daily throughout the transfer appliance request, ship, return ship, and data upload processes.



Tip

You can use the Console or the Data Transfer Utility to monitor transfer appliance-related status.

Monitoring Transfer Appliance Status

To monitor the status of your transfer appliance using the Console

1. Open the navigation menu. Under **Core Infrastructure**, go to **Object Storage** and click **Data Transfer**.
2. Find and select the transfer job for which you want to monitor transfer appliance status.
3. Under **Transfer Appliances**, look at the **Status** field.

To monitor the status of your transfer appliance using the Data Transfer Utility

At the command prompt on the host, run `dts appliance show` to monitor a transfer appliance status.

```
dts appliance show --job-id <job_id> -appliance <label>
```

Transfer Appliance Status Values

Here are the transfer appliance status values, listed in alphabetic order:

CANCELED

You can change your mind about uploading your data to Oracle Cloud Infrastructure Object Storage and cancel your transfer appliance. Ship the appliance back to Oracle and then cancel the appliance. Oracle always uses secure wipe tools on the boot and data areas whenever a transfer appliance is returned.

COMPLETE

Oracle completed your transfer appliance data upload. Your data is available in your designated bucket in Oracle Cloud Infrastructure Object Storage.

CUSTOMER LOST

You have not returned a data transfer appliance within the required 90 days.

DELIVERED

Oracle received a delivery confirmation from the shipping vendor that your transfer appliance was delivered. When the appliance is delivered, Oracle provides the date and time the appliance was received in the transfer appliance details. Appliance usage tracking begins.

ERROR

Oracle encountered an unrecoverable error trying to process your transfer appliance. Oracle cannot upload your data from the appliance. To protect your data, Oracle uses secure wipe tools on the boot and data areas any transfer appliance that cannot be processed.
Complete another request for a transfer appliance.

ORACLE PREPARING

Oracle approved your transfer appliance request. The status displays preparing until the transfer appliance is shipped to you.

ORACLE RECEIVED

Oracle received your transfer appliance shipment. The status displays Oracle received until Oracle begins processing and uploading your transfer appliance.

ORACLE RECEIVED CANCELED

You canceled your transfer appliance after you shipped the appliance back to Oracle. Oracle received your canceled transfer appliance. Oracle does *not* upload the appliance data.

PREPARING

You activated your transfer appliance. You can now copy your data onto the transfer appliance. The status displays preparing until you ship the transfer appliance back to Oracle.

PROCESSING

Oracle is processing and uploading the data from your transfer appliance. The status displays the processing status until Oracle completes uploading your data from your transfer appliance.

REJECTED

Oracle denied your transfer appliance request.



Important

If your appliance request is denied and you have questions, contact your Sales Representative or file a Service Request (SR).

REQUESTED

You successfully completed your request for a transfer appliance. The status displays requested until Oracle approves your transfer appliance request.

RETURN SHIPPED

Oracle received confirmation from the shipping vendor that you shipped your transfer appliance back to Oracle. The status displays return shipped until Oracle receives your transfer appliance.

RETURN SHIPPED CANCELED

You canceled your transfer appliance after the appliance was delivered to you or after you shipped the appliance back to Oracle. Oracle received confirmation from the shipping

vendor that your canceled transfer appliance is on the way back to Oracle. The status displays return shipped canceled until Oracle receives your transfer appliance.

SHIPPING

Oracle completed the necessary preparations and shipped your transfer appliance. When the appliance is shipped, Oracle provides the serial number of the appliance, the shipping vendor, and the tracking number in the appliance details. The status displays shipping until the appliance is delivered to you.

CHAPTER 9 Database

This chapter explains how to launch a DB System and manage databases on DB Systems.

Overview of the Database Service

The Database service offers autonomous and user-managed Oracle Database solutions. Autonomous databases are preconfigured, fully-managed environments that are suitable for either transaction processing or for data warehouse workloads. User-managed solutions are bare metal, virtual machine, and Exadata DB systems that you can customize with the resources and settings that meet your needs.

You can quickly provision a user-managed DB system or autonomous database. You have full access to the features and operations available with the database, but Oracle owns and manages the infrastructure.

For details about each offering, start with the following overview topics:

DB Systems

- Exadata DB Systems
- Bare Metal and Virtual Machine DB Systems

Autonomous Databases

The Database service offers Oracle's Autonomous Database with transaction processing and data warehouse workload types.

License Types and Bring Your Own License (BYOL) Availability

Oracle Cloud Infrastructure supports a licensing model with two license types. With **License included**, the cost of the cloud service includes a license for the Database service. With **Bring Your Own License (BYOL)**, Oracle Database customers can use existing licenses with Oracle Cloud Infrastructure. Note that Oracle Database customers remain responsible for

complying with license restrictions applicable to their BYOL licenses, as defined in their program order for those licenses.

You do not need separate on-premises licenses and cloud licenses. BYOL databases support all advanced Database service manageability functionality, including backing up and restoring a DB system, patching, and Oracle Data Guard.

You can choose BYOL when you launch an Oracle Cloud Infrastructure database or DB system. Choosing BYOL impacts how the usage data for the instance is metered and subsequent billing.

Note that on some provisioning dialogs in the Console, the BYOL option is labeled **My Organization Already Owns Oracle Database Software Licenses**.

For additional information about license pricing and features, see https://cloud.oracle.com/database/pricing.

Resource Identifiers

Most types of Oracle Cloud Infrastructure resources have a unique, Oracle-assigned identifier called an Oracle Cloud ID (OCID). For information about the OCID format and other ways to identify your resources, see Resource Identifiers.

Ways to Access Oracle Cloud Infrastructure

You can access Oracle Cloud Infrastructure using the Console (a browser-based interface) or the REST API. Instructions for the Console and API are included in topics throughout this guide. For a list of available SDKs, see Software Development Kits and Command Line Interface.

To access the Console, you must use a supported browser. Oracle Cloud Infrastructure supports the latest desktop versions of Google Chrome, Microsoft Edge, Internet Explorer 11, Safari, Firefox, and Firefox ESR. Note that private browsing mode is not supported for Firefox, Internet Explorer, or Edge. Mobile browsers are not supported.

For more information on tenancies and compartments, see "Key Concepts and Terminology" in the *Oracle Cloud Infrastructure Getting Started Guide*. For general information about using the API, see REST APIs.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to write policies that provide stricter access to database resources, see Details for the Database Service.

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

For common policies used to authorize Oracle Cloud Infrastructure Database users, see Common Policies.

For in-depth information on granting users permissions for the Database service, see Details for the Database Service in the IAM policy reference.

Limits on the Database Service

See Service Limits for a list of applicable limits and instructions for requesting a limit increase.

Many Database API operations are subject to throttling.

Exadata DB Systems

Exadata DB systems allow you to leverage the power of Exadata within the Oracle Cloud Infrastructure. An Exadata DB system consists of a quarter rack, half rack, or full rack of compute nodes and storage servers, tied together by a high-speed, low-latency InfiniBand network and intelligent Exadata software. You can configure automatic backups, optimize for different workloads, and scale up the system to meet increased demands.



Note

Exadata DB systems launched on or after March 14, 2019 run Oracle Linux 7 (OL7). Previously launched systems are running Oracle Linux 6 (OL6). See OS Updates for important information about updating existing Exadata DB system operating systems.

Supported Database Edition and Versions

Exadata DB systems require Enterprise Edition - Extreme Performance. This edition provides all the features of Oracle Database Enterprise Edition, plus all the database enterprise management packs and all the Enterprise Edition options, such as Oracle Database In-Memory and Oracle Real Application Clusters (RAC).

Exadata DB systems support the following software releases:

- Oracle Database 18c Release 1 (18.0)
- Oracle Database 12c Release 2 (12.2)
- Oracle Database 12c Release 1 (12.1)
- Oracle Database 11g Release 2 (11.2)

Subscription Types

The only subscription type available for Exadata DB systems is the Monthly Flex purchase model under Universal Credit Pricing. See https://www.oracle.com/cloud/bring-your-own-license/faq/universal-credit-pricing.html for more information.

Metering Frequency

For each Exadata DB system you provision, you are billed for the infrastructure for the first month, and then by the hour after that. Each OCPU you add to the system is billed by the hour from the time you add it.

Scaling an Exadata DB System

Two kinds of scaling operations are supported for an Exadata DB system:

- Scaling within an Exadata DB system lets you modify compute node processing power within the system.
- Scaling across Exadata DB system configurations lets you move to a different configuration, for example, from a quarter rack to a half rack.

Scaling Within an Exadata System

If an Exadata DB system requires more compute node processing power, you can scale up the number of enabled CPU cores in the system. For a non-metered Exadata DB system, you can temporarily modify the compute node processing power (bursting) or add compute node processing power on a more permanent basis. For a metered Exadata DB system, you can simply modify the number of enabled CPU cores.

You can provision an X7 Exadata DB system with zero CPU cores, or scale the DB system down to zero cores after you provision it. With zero cores, you are billed only for the infrastructure until you scale up the system. For detailed information about pricing, see https://cloud.oracle.com/iaas/database/exadata/pricing.

For information on CPU cores per configuration, see System Configuration. To learn how to scale a system, see To scale an Exadata DB system.

Scaling Across Exadata DB System Configurations

Scaling across Exadata DB system configurations enables you to move to a different system configuration. This is useful when a database deployment requires:

- Processing power that is beyond the capacity of the current system configuration.
- Storage capacity that is beyond the capacity of the current system configuration.
- A performance boost that can be delivered by increasing the number of available compute nodes.
- A performance boost that can be delivered by increasing the number of available Exadata Storage Servers.

Scaling from a quarter rack to a half rack, or from a half rack to a full rack, requires that the data associated with your database deployment is backed up and restored on a different Exadata DB system, which requires planning and coordination between you and Oracle. To start the process, submit a service request to Oracle.

System Configuration

Exadata DB systems are offered in quarter rack, half rack or full rack configurations, and each configuration consists of compute nodes and storage servers. The compute nodes are each configured with a Virtual Machine (VM). You have root privilege for the compute node VMs, so you can load and run additional software on them. However, you do not have administrative access to the Exadata infrastructure components, including the physical compute node hardware, network switches, power distribution units (PDUs), integrated lights-out management (ILOM) interfaces, or the Exadata Storage Servers, which are all administered by Oracle.

You have full administrative privileges for your databases, and you can connect to your databases by using Oracle Net Services from outside the Oracle Cloud Infrastructure. You are responsible for database administration tasks such as creating tablespaces and managing

database users. You can also customize the default automated maintenance set up, and you control the recovery process in the event of a database failure.

The following two tables provide the details for each shape's configuration.

Exadata X7 Shapes

Property	Quarter Rack	Half Rack	Full Rack
Shape Name	Exadata.Quarter2.92	Exadata.Half2.184	Exadata.Full2.368
Number of Compute Nodes	2	4	8
Total Minimum Number of Enabled CPU Cores	0	0	0
Total Maximum Number of Enabled CPU Cores	92	184	368
Total RAM Capacity	1440 GB	2880 GB	5760 GB
Number of Exadata Storage Servers	3	6	12
Total Raw Flash Storage Capacity	76.8 TB	153.6 TB	307.2 TB
Total Usable Storage Capacity	106 TB	212 TB	424 TB

Exadata X7 shapes provide 1 TB of user disk space for database homes.

Exadata X6 Shapes

Property	Quarter Rack	Half Rack	Full Rack
Shape Name	Exadata.Quarter1.84	Exadata.Half1.168	Exadata.Full1.336
Number of Compute Nodes	2	4	8
Total Minimum (Default) Number of Enabled CPU Cores	22	44	88
Total Maximum Number of Enabled CPU Cores	84	168	336
Total RAM Capacity	1440 GB	2880 GB	5760 GB
Number of Exadata Storage Servers	3	6	12
Total Raw Flash Storage Capacity	38.4 TB	76.8 TB	153.6 TB
Total Usable Storage Capacity	84 TB	168 TB	336 TB

Exadata X6 shapes provide 200 GB of user disk space for database homes.

Storage Configuration

When you launch an Exadata DB system, the storage space inside the Exadata storage servers is configured for use by Oracle Automatic Storage Management (ASM). By default, the following ASM disk groups are created:

- The DATA disk group is intended for the storage of Oracle Database data files.
- The RECO disk group is primarily used for storing the Fast Recovery Area (FRA), which is an area of storage where Oracle Database can create and manage various files related to backup and recovery, such as RMAN backups and archived redo log files.
- The DBFS and ACFS disk groups are system disk groups that support various operational purposes. The DBFS disk group is primarily used to store the shared clusterware files (Oracle Cluster Registry and voting disks), while the ACFS disk groups are primarily used to store Oracle Database binaries. Compared to the DATA and RECO disk groups, the system disk groups are so small that they are typically ignored when discussing the overall storage capacity. You should not store Oracle Database data files or backups inside the system disk groups.

The disk group names contain a short identifier string that is associated with your Exadata Database machine environment. For example, the identifier could be C2, in which case the DATA disk group would be named DATAC2, the RECO disk group would be named RECOC2, and so on.

In addition, you can create a SPARSE disk group. A SPARSE disk group is required to support Exadata snapshots. Exadata snapshots enable space-efficient clones of Oracle databases that can be created and destroyed very quickly and easily. Snapshot clones are often used for development, testing, or other purposes that require a transient database.

Impact of Configuration Settings on Storage

If you choose to perform database backups to the Exadata storage, or to create a sparse disk group, or to do both, your choices profoundly affect how storage space in the Exadata storage servers is allocated to the ASM and sparse disk groups.

The table that follows shows the approximate percentages of storage allocated for DATA, RECO, and SPARSE disk groups for each possible configuration.

Configuration Settings	DATA Disk Group	RECO Disk Group	SPARSE Disk Group
Database backups on Exadata storage: No **Sparse disk group: No**	80 %	20 %	0 %
Database backups on Exadata storage: Yes **Sparse disk group: No**	40 %	60 %	0 %
Database backups on Exadata storage: No **Sparse disk group: Yes**	60 %	20 %	20 %
Database backups on Exadata storage: Yes **Sparse disk group: Yes**	35 %	50 %	15 %

Network Setup for Exadata DB Systems

Before you set up an Exadata DB system, you must set up a virtual cloud network (VCN) and other Networking service components. This topic describes the recommended configuration for the VCN and several related requirements for the Exadata DB system.

VCN and Subnets

To launch an Exadata DB system, you must have:

- A VCN in the region where you want the DB system
- At least two subnets in the VCN. The two subnets are:

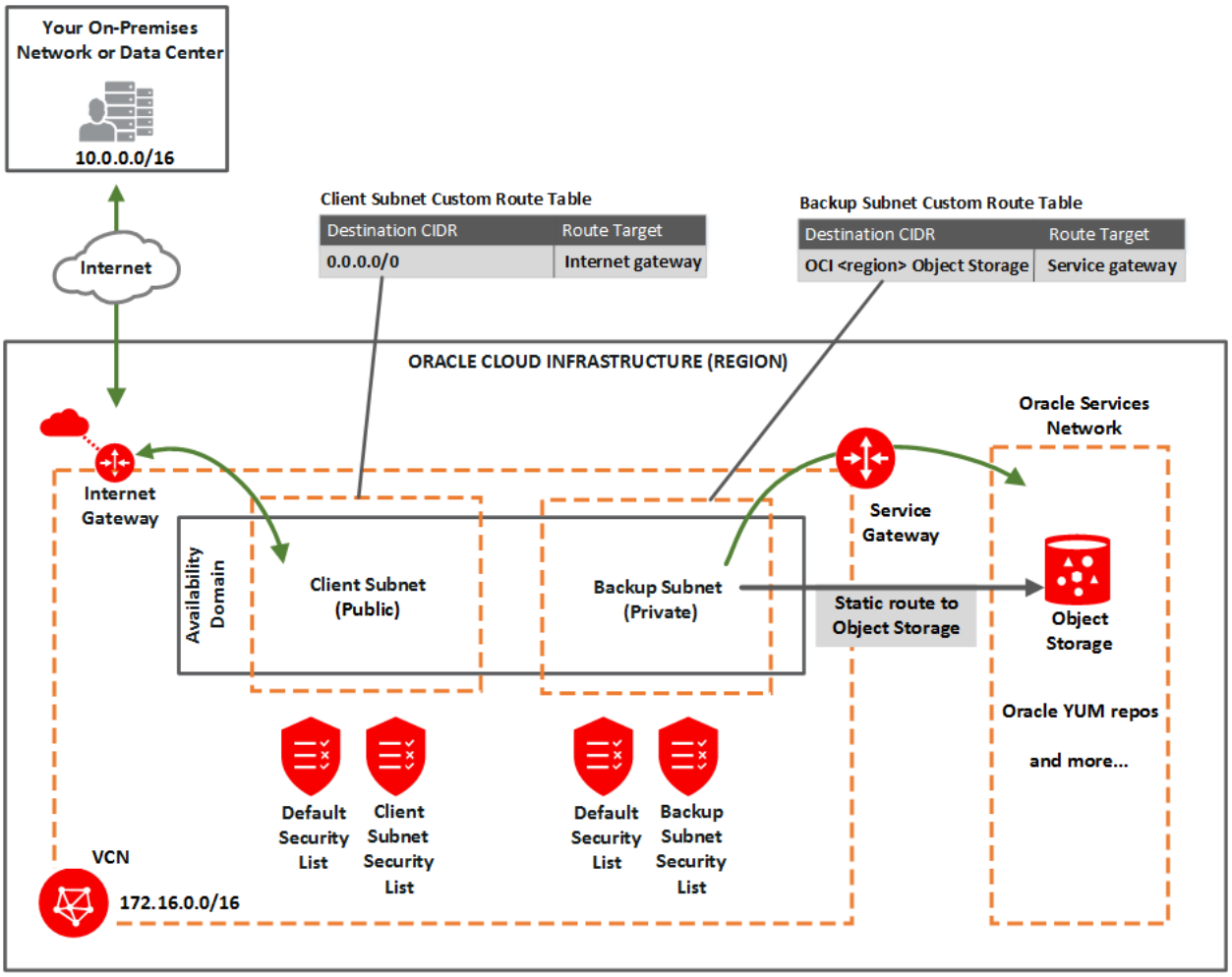
- Client subnet
- Backup subnet

In general, Oracle recommends using regional subnets, which span all availability domains in the region. If you instead use AD-specific subnets, both the client and backup subnets must be in the same availability domain. The important thing to know for your DB system is that the resources you create in the two subnets must be in the same availability domain. For more information, see About Regional Subnets.

You will create custom route tables and security lists for each subnet. More information follows about that.

Option 1: Public Client Subnet with Internet Gateway

This option can be useful when doing a proof-of-concept or development work. You can use this setup in production if you want to use an internet gateway with the VCN, or if you have services that run only on a public network and need access to the database. See the following diagram and description.



You set up:

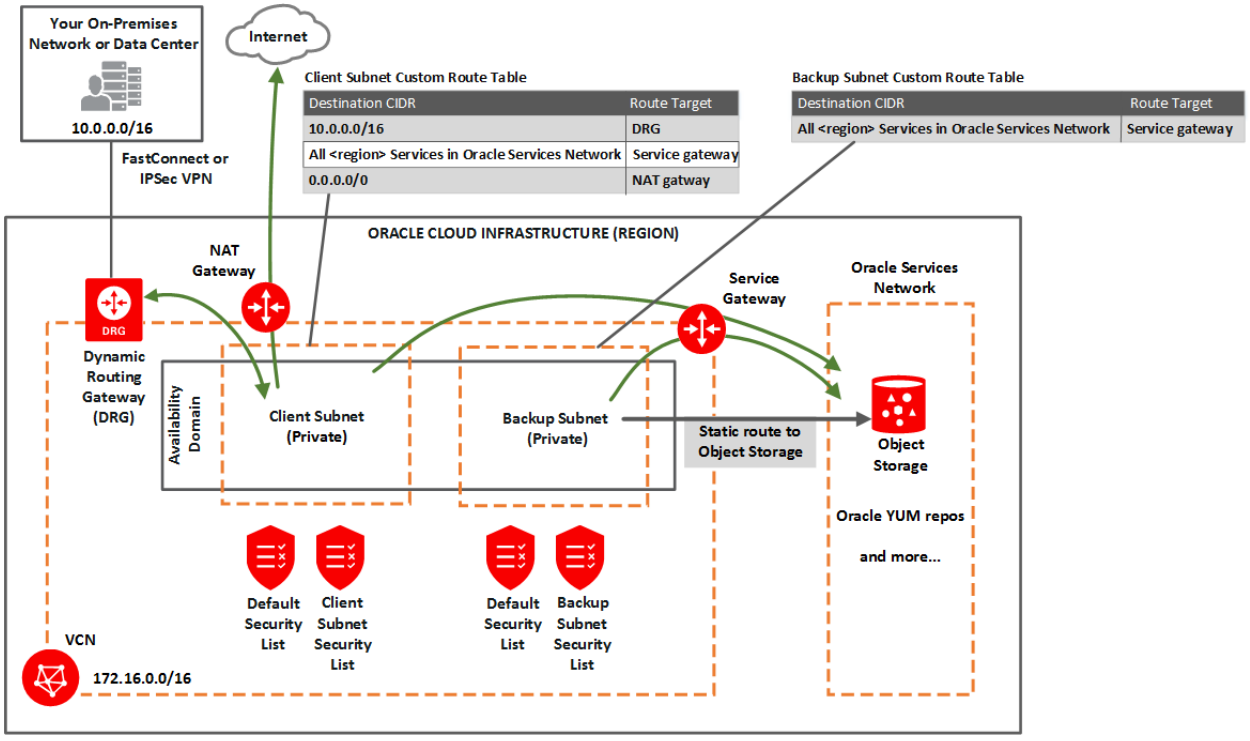
- Subnets:
  - *Public* client subnet (*public* means that the resources in the subnet can have public IP addresses at your discretion).

    - Private backup subnet (*private* means that the resources in the subnet cannot have public IP addresses and therefore cannot receive incoming connections from the internet).
- Gateways for the VCN:
    - Internet gateway (for use by the client subnet).
    - Service gateway (for use by the backup subnet). Also see Option 1: Service Gateway Access Only to Object Storage.
- Route tables:
    - Custom route table for the public client subnet, with a route for 0.0.0.0/0, and target = the internet gateway.
    - Separate custom route table for the private backup subnet, with a route rule for the service CIDR label called **OCI *<region>* Object Storage**, and target = the service gateway. Also see Option 1: Service Gateway Access Only to Object Storage.
- Security lists:
    - Default security list (with its default rules) for both the public client subnet and private backup subnet. **This security list contains several basic rules that are necessary for the DB system (such as ingress SSH access to the DB system, and general egress from the DB system).**
    - Custom security list for the public client subnet only. For the rules to use, see Rules for Client Subnet's Custom Security List.
    - Separate custom security list for the private backup subnet only. For the rules to use, see Rule for Backup Subnet's Custom Security List.
- Static route on the DB system's compute nodes (to enable access to Object Storage by way of the backup subnet).

> **Important**
>
> See this known issue for information about configuring route rules with *service gateway* as the target on route tables associated with public subnets.

OPTION 2: PRIVATE SUBNETS

This option is recommended for a production system. Both subnets are private and cannot be reached from the internet. See the following diagram and description.



You set up:

- Subnets:
  - *Private* client subnet.
  - Private backup subnet.
- Gateways for the VCN:
  - Dynamic routing gateway (DRG), with a FastConnect or IPSec VPN to your on-premises network (for use by the client subnet).
  - Service gateway (for use by the backup subnet to reach Object Storage, and for use by the client subnet to reach the Oracle YUM repo for OS updates). Also see Option 2: Service Gateway Access to Both Object Storage and YUM Repos.
  - NAT gateway (for use by the client subnet to reach public endpoints not supported by the service gateway).
- Route tables:
  - Custom route table for the private client subnet, with two rules:
    - A rule for the on-premises network's CIDR, and target = DRG.
    - A rule for the service CIDR label called **All *<region>* Services in Oracle Services Network**, and target = the service gateway. The *Oracle Services Network* is a conceptual network in Oracle Cloud Infrastructure that is reserved for Oracle services. The rule enables the client subnet to reach the regional Oracle YUM repo for OS updates. Also see Option 2: Service Gateway Access to Both Object Storage and YUM Repos.
    - A rule for 0.0.0.0/0, and target = NAT gateway.
  - Separate custom route table for the private backup subnet, with one rule:
    - The same rule as for the client subnet: for the service CIDR label called **All *<region>* Services in Oracle Services Network**, and target = the service gateway. This rule enables the backup subnet to reach the regional Object Storage for backups.
- Security lists:

- Default security list (with its default rules) for both the private client subnet and private backup subnet. **This security list contains several basic rules that are necessary for the DB system (such as ingress SSH access to the DB system, and general egress from the DB system).**
- Custom security list for the private client subnet only. For the rules to use, see Rules for Client Subnet's Custom Security List.
- Separate custom security list for the private backup subnet only. For the rules to use, see Rule for Backup Subnet's Custom Security List.

- Static route on the DB system's compute nodes (to enable access to Object Storage by way of the backup subnet).

Requirements for IP Address Space

If you're setting up Exadata DB systems (and thus VCNs) in more than one region, make sure the IP address space of the VCNs does not overlap. This is important if you want to set up disaster recovery with Oracle Data Guard.

The two subnets you create for the Exadata DB system must not overlap with 192.168.128.0/20.

The following table lists the *minimum* required subnet sizes, depending on the Exadata rack size. For the client subnet, each node requires two IP addresses, and in addition, three addresses are reserved for Single Client Access Names (SCANs). For the backup subnet, each node requires one address.



Tip

The Networking service reserves three IP addresses in each subnet. Allocating a larger space for the subnet than the minimum required (for example, at least /25 instead of /28) can reduce the relative impact of those reserved addresses on the subnet's available space.

Rack Size	Client Subnet: # Required IP Addresses	Client Subnet: Minimum Size	Backup Subnet: # Required IP Addresses	Backup Subnet: Minimum Size
Quarter	(2 addresses * 2 nodes) + 3 for SCANs + 3 reserved in subnet = 10	/28 (16 IP addresses)	(1 address * 2 nodes) + 3 reserved in subnet = 5	/29 (8 IP addresses)
Half	(2 * 4 nodes) + 3 + 3 = 14	/28 (16 IP addresses)	(1 * 4 nodes) + 3 = 7	/29 (8 IP addresses)
Full	(2* 8 nodes) + 3 + 3 = 22	/27 (32 IP addresses)	(1 * 8 nodes) + 3 = 11	/28 (16 IP addresses)

VCN Creation Wizard: Not for Production

The Networking section of the Console includes a handy wizard that creates a VCN along with related resources. It can be useful if you just want to try launching an instance. However, the wizard automatically chooses the address ranges and creates public subnets and an internet gateway. You may not want this for your production network, so Oracle recommends you create the VCN and other resources individually yourself instead of using the wizard.

DNS: Short Names for the VCN, Subnets, and DB System

For the nodes to communicate, the VCN must use the Internet and VCN Resolver. It enables hostname assignment to the nodes, and DNS resolution of those hostnames by resources in the VCN. It enables round robin resolution of the database's SCANs. It also enables resolution of important service endpoints required for backing up databases, patching, and updating the cloud tooling on an Exadata DB system. The Internet and VCN Resolver is the VCN's default choice for DNS in the VCN. For more information, see DNS in Your Virtual Cloud Network and also DHCP Options.

When you create the VCN, subnets, and Exadata, you must carefully set the following identifiers, which are related to DNS in the VCN:

- VCN domain label
- Subnet domain label
- Hostname prefix for the Exadata DB system

These values make up the node's fully qualified domain name (FQDN):

<hostname_prefix>-######.*<subnet_domain_label>*.*<vcn_domain_label>*.oraclevcn.com

For example:

exacs-abcde1.clientpvtad1.acmevcniad.oraclevcn.com

In this example, you assign `exacs` as the hostname prefix when you create the Exadata DB system. The Database service automatically appends a hyphen and a five-letter string with the node number at the end. For example:

- Node 1: `exacs-abcde1.clientpvtad1.acmevcniad.oraclevcn.com`
- Node 2: `exacs-abcde2.clientpvtad1.acmevcniad.oraclevcn.com`
- Node 3: `exacs-abcde3.clientpvtad1.acmevcniad.oraclevcn.com`
- And so on

Requirements for the hostname prefix:

- Maximum 12 characters
- Cannot be the string *localhost*

Requirements for the VCN and subnet domain labels:

- Recommended maximum: 14 characters each. The actual underlying requirement is a total of 28 characters *across both domain labels* (excluding the period between the labels). For example, both of these are acceptable: `subnetad1.verylongvcnphx` or `verylongsubnetad1.vcnphx`. For simplicity, the recommendation is 14 characters each.
- No hyphens.

- Recommended: include the region name in the VCN's domain label, and include the availability domain name in the subnet's domain label.

<12_chars_max>-######.*<14_chars_max>*.*<14_chars_max>*.oraclevcn.com

In general, the FQDN has a maximum total limit of 63 characters.

The preceding maximums are not enforced when you create the VCN and subnets. However, if the labels exceed the maximum, the Exadata deployment fails.

DNS: Between On-Premises Network and VCN

To enable the use of hostnames when on-premises hosts and VCN resources communicate with each other, you have two options:

- Set up an instance in the VCN to be a custom DNS server. For an example of an implementation of this scenario with the Oracle Terraform provider, see Hybrid DNS Configuration.
- Manage hostname resolution yourself manually.

Node Access to Object Storage: Static Route

Access to Oracle Cloud Infrastructure Object Storage is required for backing up databases, patching, and updating the cloud tooling on an Exadata DB system. Regardless of how you set up the VCN with that access (for example, with a service gateway), you must configure a static route to Object Storage on each of the compute nodes in the cluster. This is required because, by default, all traffic in an Exadata DB system is routed through the data network. You need the traffic destined for Object Storage to be routed instead through the backup interface (BONDETH1).



Important

You must configure a static route for Object Storage access *on each compute node* in an Exadata DB system. Otherwise, attempts to back up databases, patch, or update tooling on the system might fail.

Object Storage IP allocations

Oracle Cloud Infrastructure Object Storage uses the CIDR block IP range 134.70.0.0/17 for all regions. This range was introduced in April and May of 2018.

As of June 1, 2018, Object Storage no longer supports the following discontinued IP ranges. Oracle recommends that you remove these older IP addresses from your access-control lists, firewall rules, and other rules after you have adopted the new IP ranges.

The **discontinued** IP ranges are:

- eu-frankfurt-1: 130.61.0.0/16
- uk-london-1: 132.145.0.0/16
- us-ashburn-1: 129.213.0.0/16
- us-phoenix-1: 129.146.0.0/16

To configure a static route for Object Storage access

1. SSH to a compute node in the Exadata DB system.

   ```
   ssh -i <private_key_path> opc@<node_ip_address>
   ```

2. Log in as opc and then sudo to the root user. Use `sudo su -` with a hyphen to invoke the root user's profile.

   ```
   login as: opc

   [opc@dbsys ~]$ sudo su -
   ```

3. Identify the gateway configured for the BONDETH1 interface.

   ```
   [root@dbsys ~]# grep GATEWAY /etc/sysconfig/network-scripts/ifcfg-bondeth1 |awk -F"=" '{print
   $2}'


   10.0.4.1
   ```

4. Add the following static rule for BONDETH1 to the `/etc/sysconfig/network-scripts/route-bondeth1` file:

```
10.0.X.0/XX dev bondeth1 table 211
default via <gateway> dev bondeth1 table 211
134.70.0.0/17 via <gateway_from_previous_step> dev bondeth1
```

5. Restart the interface.

   ```
   [root@dbsys ~]# ifdown bondeth1; ifup bondeth1;
   ```

   The file changes from the previous step take effect immediately after the ifdown and ifup commands run.

6. Repeat the preceding steps on *each* compute node in the Exadata DB system.

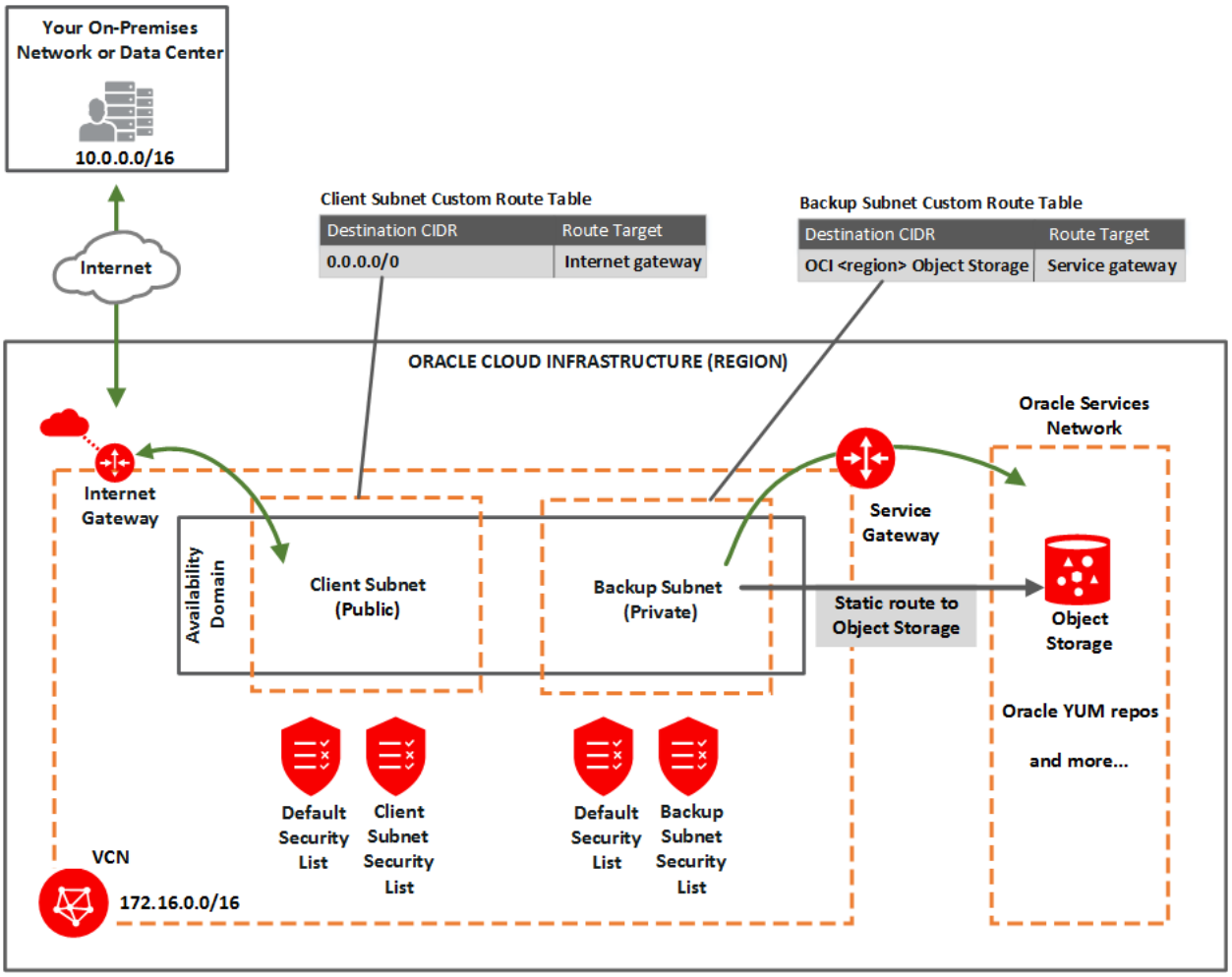
Service Gateway for the VCN

Your VCN needs access to both Object Storage for backups and Oracle YUM repos for OS updates.

Depending on whether you use option 1 or option 2 described previously, you use the service gateway in different ways. See the next two sections.

Option 1: Service Gateway Access Only to Object Storage

You configure the *backup subnet* to use the service gateway for access only to Object Storage. As a reminder, here's the diagram for option 1:



In general, you must:

- Perform the tasks for setting up a service gateway on a VCN, and specifically enable the service CIDR label called **OCI *<region>* Object Storage**.
- In the task for updating routing, add a route rule to the *backup* subnet's custom route table. For the destination service, use **OCI *<region>* Object Storage** and target = the service gateway.

- In the optional task for updating security lists in the subnet, perform the task on the *backup* subnet's custom security list. Set up a security list rule with the destination service set to **OCI *<region>* Object Storage**. See Rule for Backup Subnet's Custom Security List.

Option 2: Service Gateway Access to Both Object Storage and YUM Repos

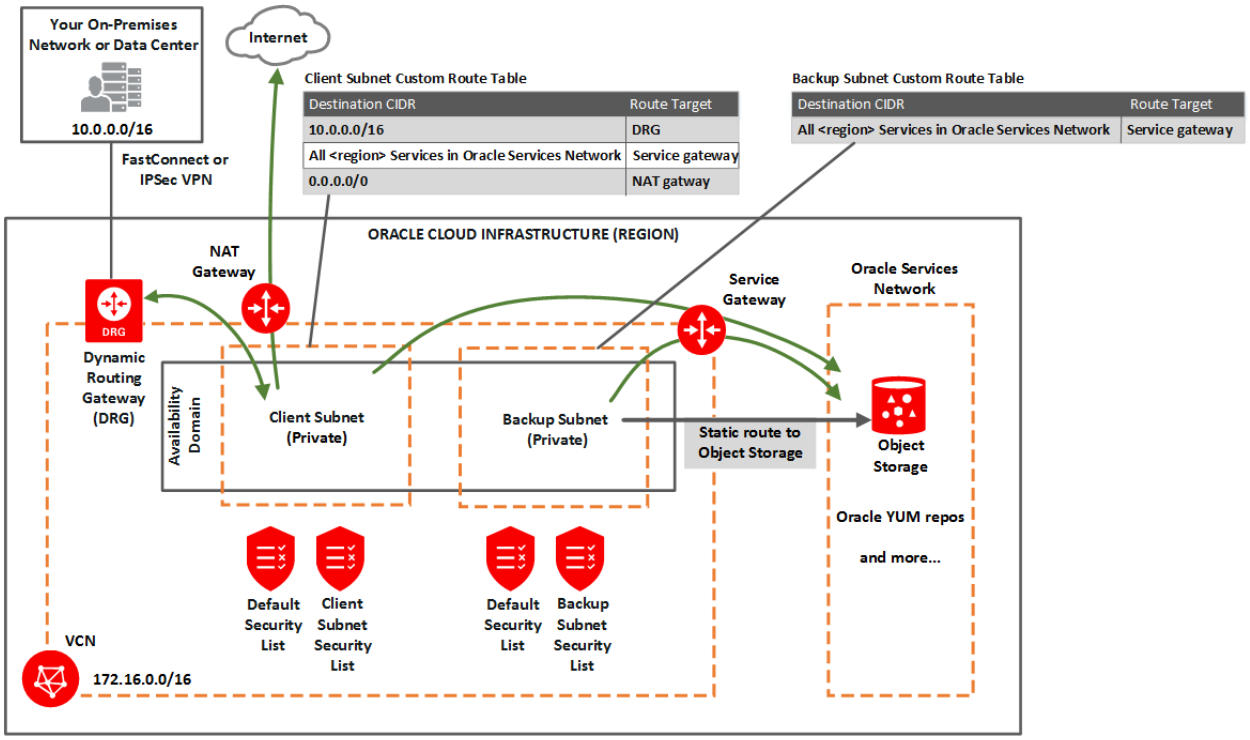
You configure *both the client subnet and backup subnet* to use the service gateway for access to the Oracle Services Network, which includes both Object Storage and the Oracle YUM repos.



Important

See this known issue for information about accessing Oracle YUM services through the service gateway.

As a reminder, here's the diagram for option 2:



In general, you must:

- Perform the tasks for setting up a service gateway on a VCN, and specifically enable the service CIDR label called **All *<region>* Services in Oracle Services Network**.
- In the task for updating routing in each subnet, add a rule to each subnet's custom route table. For the destination service, use **All *<region>* Services in Oracle Services Network** and target = the service gateway.
- In the optional task for updating security lists in the subnet, perform the task on the *backup* subnet's custom security list. Set up a security list rule with the destination service set to **OCI *<region>* Object Storage**. See Rule for Backup Subnet's Custom Security List. Note that the client subnet already has a broad egress rule that covers access to the YUM repos.

Here are a few additional details about using the service gateway for option 2:

- Both the client subnet and backup subnet use the service gateway, but to access different services. You cannot enable both the **OCI *<region>* Object Storage** service CIDR label and the **All *<region>* Services in Oracle Services Network** for the service gateway. To cover the needs of both subnets, you must enable **All *<region>* Services in Oracle Services Network** for the service gateway. The VCN can have only a single service gateway.
- Any route rule that targets a given service gateway must use an enabled service CIDR label and not a CIDR block as the destination for the rule. That means for option 2, the route tables for both subnets must use **All *<region>* Services in Oracle Services Network** for their service gateway rules.
- Unlike route rules, security list rules can use either *any* service CIDR label (whether the VCN has a service gateway or not) or a CIDR block as the source or destination CIDR for the rule. Therefore, although the backup subnet has a route rule that uses **All *<region>* Services in Oracle Services Network**, the subnet can have a security list rule that uses **OCI *<region>* Object Storage**. See Rule for Backup Subnet's Custom Security List.

Security Lists for the Subnets

Oracle recommends that you use at least two security lists with each of the subnets.

For the client subnet:

- VCN's default security list (with all its default rules). **This security list contains several basic rules that are necessary for the DB system (such as ingress SSH access to the DB system, and general egress from the DB system).**
- Custom security list specifically for the client subnet (see Rules for Client Subnet's Custom Security List).

For the backup subnet:

- VCN's default security list (with all its default rules).
- Custom security list specifically for the backup subnet (see Rule for Backup Subnet's Custom Security List).

You must create the two custom security lists yourself. The VCN's default security list is automatically created when you create the VCN. When you create a subnet, you can choose which security lists the subnet will use, or you can choose the security lists after the subnet is created.



Warning

Do not remove the default egress rule from the default security list. If you do, make sure to instead include a replacement egress rule in the client subnet's security list, like so:

- **Stateless:** No (all rules must be stateful)
- **Destination Type:** CIDR
- **Destination CIDR:** 0.0.0.0/0
- **IP Protocol:** All

Rules for Client Subnet's Custom Security List

Create the following security list rules in the client subnet's custom security list and associate the security list with the client subnet.

Ingress rule 1: Allows ONS and FAN traffic from within the client subnet

The first rule is recommended and enables the Oracle Notification Services (ONS) to communicate about Fast Application Notification (FAN) events.

- **Stateless:** No (all rules must be stateful)
- **Source Type:** CIDR

- **Source CIDR:** Client subnet's CIDR
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range:** 6200

Ingress rule 2: Allows SQL*NET traffic from within the client subnet

This rule is for SQL*NET traffic and is required in these cases:

- If you need to enable client connections to the database
- If you plan to use Oracle Data Guard

- **Stateless:** No (all rules must be stateful)
- **Source Type:** CIDR
- **Source CIDR:** Client subnet's CIDR
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range:** 1521



Important

The two preceding rules only cover connections initiated from within the client subnet. If you have a client that resides *outside the VCN*, Oracle recommends setting up two *additional* similar rules that instead have the **Source CIDR** set to the public IP address of the client.

The next four rules (two ingress, two egress) allow TCP and ICMP traffic inside the client subnet and enable the nodes to communicate with each other. If TCP connectivity fails across the nodes, the Exadata DB system fails to provision.

Ingress rule 3: Allows all TCP traffic inside the client subnet

- **Stateless:** No (all rules must be stateful)
- **Source Type:** CIDR
- **Source CIDR:** Client subnet's CIDR
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range:** All

Ingress rule 4: Allows all ICMP traffic inside the client subnet

- **Stateless:** No (all rules must be stateful)
- **Source Type:** CIDR
- **Source CIDR:** Client subnet's CIDR
- **IP Protocol:** ICMP
- **Type and Code:** All

Egress rule 1: Allows all TCP traffic inside the client subnet

- **Stateless:** No (all rules must be stateful)
- **Destination Type:** CIDR
- **Destination CIDR:** Client subnet's CIDR
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range:** All

Egress rule 2: Allows all ICMP traffic inside the client subnet

- **Stateless:** No (all rules must be stateful)
- **Destination Type:** CIDR
- **Destination CIDR:** Client subnet's CIDR
- **IP Protocol:** ICMP
- **Type and Code:** All

The next egress rule is redundant with the default egress rule in the default security list. It is optional but recommended in case the default security list's egress rule is inadvertently changed. The rule is important because it allows connections to the Oracle YUM repos.

Egress rule 3: Allows all egress traffic

- **Stateless:** No (all rules must be stateful)
- **Destination Type:** CIDR
- **Destination CIDR:** 0.0.0.0/0
- **IP Protocol:** All

Rule for Backup Subnet's Custom Security List

The following rule enables the DB system to communicate with Object Storage through the service gateway. It is redundant with the default egress rule in the default security list. It is optional but recommended in case the default security list's rules are inadvertently changed.

Put this rule in the backup subnet's custom security list and associate the security list with the backup subnet.

Egress rule: Allows access to Object Storage

- **Stateless:** No (all rules must be stateful)
- **Destination Type:** Service

- **Destination Service:** The service CIDR label called **OCI *<region>* Object Storage**
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range:** 443 (HTTPS)

Managing Exadata DB Systems

This topic explains how to launch, start, stop, terminate, scale, manage licenses for, and check the status of an Exadata DB system. It also describes how to configure required access to the Oracle Cloud Infrastructure Object Storage service and set up DNS.

When you launch an Exadata DB system using the Console or the API, the system is provisioned to support Oracle databases. The service creates an initial database based on the options you provide and some default options described later in this topic.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let database admins manage database systems lets the specified group do everything with databases and related Database resources.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for databases, see Details for the Database Service.

Prerequisites

- The public key, in OpenSSH format, from the key pair that you plan to use for connecting to the DB System via SSH. A sample public key, abbreviated for readability, is shown below.

  ```
  ssh-rsa AAAAB3NzaC1yc2EAAAABJQAA....lo/gKMLVM2xzc1xJr/Hc26biw3TXWGEakrK1OQ== rsa-key-20160304
  ```

  For more information, see Managing Key Pairs on Linux Instances.
- A correctly configured virtual cloud network (VCN) to launch the DB system in. Its related networking resources (gateways, route tables, security lists, DNS, and so on) must also be configured as necessary for the DB system. For more information, see Network Setup for Exadata DB Systems.

Default Options for the Initial Database

To simplify launching a DB system in the Console and when using the API, the following default options are used for the initial database.

- **Console Enabled:** False
- **Create Container Database:** False for version 11.2.0.4 databases. Otherwise, true.
- **Create Instance Only (for standby and migration):** False
- **Database Home ID:** Creates a database home
- **Database Language:** AMERICAN
- **Database Sizing Template:** odb2
- **Database Storage:** Automatic Storage Management (ASM)
- **Database Territory:** AMERICA
- **Database Unique Name:** The user-specified database name and a system-generated suffix, for example, dbtst_phx1cs.
- **PDB Admin Name:** pdbuser (Not applicable for version 11.2.0.4 databases.)

For a list of the database options that you can set in the Console, see To launch an Exadata DB system.

Using the Console

To launch an Exadata DB system

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
3. Click **Launch DB System**.
4. In the **Launch DB System** dialog, enter the following:

   DB System Information

   - **Compartment:** By default, the DB system launches in your current compartment and you can use the network resources in that compartment. Click the **click here** link in the dialog box if you want to enable compartment selection for the DB system, network, and subnet resources.
   - **Display Name:** A friendly, display name for the DB system. The name doesn't need to be unique. An Oracle Cloud Identifier (OCID) will uniquely identify the DB system.
   - **Availability Domain:** The availability domain in which the DB system resides.
   - **Shape Type:** The type of shape to use to launch the DB system. The shape type filters the list of available shapes to select from.
   - **Shape:** The shape to use to launch the DB system. The shape determines the type of DB system and the resources allocated to the system.

     ### Exadata shapes

     Exadata X6 shapes:

     - **Exadata.Quarter1.84:** Provides a 2-node Exadata DB system with 22 enabled CPU cores, with up to 62 additional CPU cores, and 84 TB of usable storage.

- **Exadata.Half1.168:** Provides a 4-node Exadata DB system with 44 enabled CPU cores, with up to 124 additional CPU cores, and 168 TB of usable storage.
- **Exadata.Full1.336:** Provides an 8-node Exadata DB system with 88 enabled CPU cores, with up to 248 additional CPU cores, and 336 TB of usable storage.

Exadata X7 shapes:

- **Exadata.Quarter2.92:** Provides a 2-node Exadata DB system with up to 92 CPU cores, and 106 TB of usable storage.
- **Exadata.Half2.184:** Provides a 4-node Exadata DB system with up to 184 CPU cores, and 212 TB of usable storage.
- **Exadata.Full2.368:** Provides an 8-node Exadata DB system with up to 368 CPU cores, and 424 TB of usable storage.

All Exadata shapes provide 720 GB RAM per node and unlimited I/O, and support only Enterprise Edition - Extreme Performance. For more details about Exadata shapes, see System Configuration.

- **Cluster Name:** A unique cluster name for a multi-node DB system. The name must begin with a letter and contain only letters (a-z and A-Z), numbers (0-9) and hyphens (-). The cluster name can be no longer than 11 characters and is not case sensitive.
- **Total Node Count:** The number of nodes in the DB system. The number depends on the shape you select.
- **Oracle Database Software Edition:** The database edition supported by the DB system. For bare metal systems, you can mix supported database releases on the DB system to include older database versions, but not editions. The database edition cannot be changed and applies to all the databases in this DB system. Virtual machine systems support only one database.
- **CPU Core Count:** The number of CPU cores for the DB system. Displays only if

you select a shape that allows you to configure the number of cores. The text below the field indicates the acceptable values for that shape. For a multi-node DB system, the core count is evenly divided across the nodes.

You can increase the CPU cores to accommodate increased demand after you launch the DB system.

For an Exadata X7 DB system, you can specify zero (0) cores when you launch the system. This will provision the system and immediately stop it. See Scaling Within an Exadata System for information about CPU core scaling and the impact on billing.

- **License Type:** The type of license you want to use for the DB system. Your choice affects metering for billing.
  - **License included** means the cost of the cloud service includes a license for the Database service.
  - **Bring Your Own License (BYOL)** means you are an Oracle Database customer with an Unlimited License Agreement or Non-Unlimited License Agreement and want to use your license with Oracle Cloud Infrastructure. This removes the need for separate on-premises licenses and cloud licenses.
- **SSH Public Key:** The public key portion of the key pair you want to use for SSH access to the DB system. To provide multiple keys, paste each key on a new line. Make sure each key is on a single, continuous line. The length of the combined keys cannot exceed 10,000 characters.
- **Storage Allocation:** The configuration settings that determine the percentage of storage assigned to DATA, RECO, and optionally, SPARSE disk:
  - **Database Backups on Exadata Storage:** Select this option if you intend to perform database backups to the local Exadata storage within your Exadata DB system environment. If you select this option, more space is allocated to the RECO disk group, which is used to store backups on Exadata storage. If you do not select this option, more space is allocated to the DATA disk group, which enables you to store more information in your

databases.

- **Create Sparse Disk Group:** Select this configuration option if you intend to use snapshot functionality within your Exadata DB system environment. If you select this option, the SPARSE disk group is created, which enables you to use Exadata DB system snapshot functionality for PDB sparse cloning. If you do not select this option, the SPARSE disk group is not created and Exadata DB system snapshot functionality will not be available on any database deployments that are created in the environment.



Important

Creating a sparse disk group impacts the storage available for the ASM disk groups (DATA and RECO) and you cannot change the storage allocation configuration after you provision your DB system. For information about the percentage of storage that will be assigned to DATA, RECO, and SPARSE disk based on your configuration, see Impact of Configuration Settings on Storage. Similar information will display under the options in the Console dialog.

- **Disk Redundancy:** *(Available when you select Show Advanced Options)* The type of redundancy configured for the DB system.
  - **Normal** is 2-way mirroring, recommended for test and development systems. This option is not available for Exadata DB systems.
  - **High** is 3-way mirroring, recommended for production systems.
- **Time Zone:** *(Available when you select Show Advanced Options)* The default time zone for the DB system is UTC, but you can specify a different time zone. The

time zone options are those supported in both the Java.util.TimeZone class and the Oracle Linux operating system. For more information, see DB System Time Zone.



Tip

If you want to set a time zone other than UTC or the browser-detected time zone and if you do not see the time zone you are interested in, try selecting "Miscellaneous" as the time zone prefix.

NETWORK INFORMATION

- **Virtual Cloud Network Compartment:** The compartment containing the network in which to launch the DB system.
- **Virtual Cloud Network:** The VCN in which to launch the DB system.
- **Subnet Compartment:** The compartment containing a subnet within the cloud network to attach the DB system to.
- **Client Subnet:** The subnet to which the Exadata DB system should attach.
  Do not use a subnet that overlaps with 192.168.128.0/20. This restriction applies to both the client subnet and backup subnet.
- **Backup Subnet:** The subnet to use for the backup network , which is typically used to transport backup information to and from Oracle Cloud Infrastructure Object Storage, and for Data Guard replication.
  Do not use a subnet that overlaps with 192.168.128.0/20. This restriction applies to both the client subnet and backup subnet.
  If you plan to back up databases to Object Storage, see the network prerequisites in Backing Up an Exadata Database.

- **Hostname Prefix:** Your choice of host name for the Exadata DB system. The host name must begin with an alphabetic character, and can contain only alphanumeric characters and hyphens (-). The maximum number of characters allowed for an Exadata DB system is 12.

  Important

  The host name must be unique within the subnet. If it is not unique, the DB system will fail to provision.

- **Host Domain Name:** The domain name for the DB system. If the selected subnet uses the Oracle-provided Internet and VCN Resolver for DNS name resolution, this field displays the domain name for the subnet and it can't be changed. Otherwise, you can provide your choice of a domain name. Hyphens (-) are not permitted.

  If you plan to store database backups in Object Storage, Oracle recommends using a VCN Resolver for DNS name resolution for the client subnet as it automatically resolves the Swift endpoints used for backups.
- **Host and Domain URL:** Combines the host and domain names to display the fully qualified domain name (FQDN) for the database. The maximum length is 64 characters.

DATABASE INFORMATION

- **Database Name:** The name for the database. The database name must begin with an alphabetic character and can contain a maximum of eight alphanumeric characters. Special characters are not permitted.
- **Database Version:** The version of the initial database created on the DB system when it is launched. After the DB system is active, you can create additional databases on it. You can mix database versions on the DB system, but not

editions.

- **PDB Name:** *Not applicable to version 11.2.0.4.* The name of the pluggable database. The PDB name must begin with an alphabetic character, and can contain a maximum of 8 alphanumeric characters. The only special character permitted is the underscore (_).
- **Database Admin Password:**
  A strong password for SYS, SYSTEM, TDE wallet, and PDB Admin. The password must be 9 to 30 characters and contain at least 2 uppercase, 2 lowercase, 2 numeric, and 2 special characters. The special characters must be _, #, or -. The password must not contain the username (SYS, SYSTEM, and so on) or the word "oracle" either in forward or reversed order and regardless of casing.
- **Confirm Database Admin Password:** Re-enter the Database Admin Password you specified.
- **Database Workload:**
  Select the workload type that best suits your application.
  - **Online Transactional Processing (OLTP)** configures the database for a transactional workload, with a bias towards high volumes of random data access.
  - **Decision Support System (DSS)** configures the database for a decision support or data warehouse workload, with a bias towards large data scanning operations.
- **Character Set:** The character set for the database. The default is AL32UTF8.
- **National Character Set:** The national character set for the database. The default is AL16UTF16.
- **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your

administrator.

5. Click **Launch DB System**.
   The DB system appears in the list with a status of Provisioning. The DB system's icon changes from yellow to green (or red to indicate errors).
6. Wait for the DB system's icon to turn green, with a status of Available, and then click the highlighted DB system name.
   Details about the DB system are displayed.



Note

An X7 Exadata DB system launched with zero cores will be in a Stopped state until you scale it up.

7. Note the IP addresses; you'll need the private or public IP address, depending on network configuration, to connect to the DB system.

To check the status of an Exadata DB system

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. In the list of DB systems, find the system you're interested in and check its icon. The color of the icon and the text below it indicates the status of the system.
   - **Provisioning:** Yellow icon. Resources are being reserved for the DB system, the system is booting, and the initial database is being created. Provisioning can take several minutes. The system is not ready to use yet.
   - **Available:** Green icon. The DB system was successfully provisioned. A few minutes after the system enters this state, you can SSH to it and begin using it.

- **Terminating:** Gray icon. The DB system is being deleted by the terminate action in the Console or API.
- **Terminated:** Gray icon. The DB system has been deleted and is no longer available.
- **Failed:** Red icon. An error condition prevented the provisioning or continued operation of the DB system.

To view the status of a database node, under Resources, click **Nodes** to see the list of nodes. In addition to the states listed for a DB system, a node's status can be one of the following:

- **Starting:** Yellow icon. The database node is being powered on by the start or reboot action in the Console or API.
- **Stopping:** Yellow icon. The database node is being powered off by the stop or reboot action in the Console or API.
- **Stopped:** Yellow icon. The database node was powered off by the stop action in the Console or API.

You can also check the status of DB systems and database nodes using the ListDbSystems or ListDbNodes API operations, which return the `lifecycleState` attribute.

To start, stop, or reboot an Exadata DB system

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. In the list of DB systems, find the DB system you want to stop or start, and then click its name to display details about it.
4. In the list of nodes, click the Actions icon (three dots) for a node and then click one of the following actions:
   - **Start:** Restarts a stopped node. After the node is restarted, the **Stop** action is enabled.

- **Stop:** Shuts down the node. After the node is powered off, the **Start** action is enabled.
- **Reboot:** Shuts down the node, and then restarts it.



Note

- For billing purposes, the **Stop** state has no effect on the resources you consume. Billing continues for nodes that you stop, and related resources continue to apply against any relevant quotas. You must **Terminate** a DB system to remove its resources from billing and quotas.
- After you restart or reboot a node, the floating IP address might take several minutes to be updated and display in the Console.

To scale an Exadata DB system

If an Exadata DB system requires more compute node processing power, you can scale up (burst) the number of enabled CPU cores in the system.

You can also scale an X7 Exadata DB system down to zero CPU cores to temporarily stop the system and be charged only for the hardware infrastructure. For more information about scaling down, see Scaling Within an Exadata System.

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. In the list of DB systems, find the system you want to scale and click its highlighted

name.
The system details are displayed.

4. Click **Scale Up/Down** and then change the number in **Total CPU Core Count**. The text below the field indicates the acceptable values, based on the shape used when the DB system was launched.
5. Click **Scale Up/Down DB System**.



Note

If you scale an X7 Exadata DB system down to zero CPU cores, the floating IP address of the nodes might take several minutes to be updated and display in the Console.

To terminate an Exadata DB system

Terminating a DB system permanently deletes it and any databases running on it.



Note

The database data is local to the DB system and is lost when the system is terminated. Oracle recommends that you back up any data in the DB system before terminating it.

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
A list of DB systems is displayed.

3. For the DB system you want to terminate, click the Actions icon (three dots), and then click **Terminate**.
4. Confirm when prompted.
   The DB system's icon indicates Terminating.

At this point, you cannot connect to the system and any open connections are terminated.

To manage your BYOL database licenses

If you want to control the number of database licenses that you run at any given time, you can scale up or down the number of OCPUs on the instance. These additional licenses are metered separately.

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. In the list of DB systems, find the system you want to scale and click its highlighted name.
   The system details are displayed.
4. Click **Scale Up/Down OCPU** and then change the number.

To manage tags for your DB systems and database resources

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. Find the DB system or database resource you're interested in, and click the name.
4. Click the **Tags** tab to view or edit the existing tags. Or click **Apply Tag(s)** to add new tags.

For more information, see Resource Tags.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage DB system components.

DB systems:

- GetDbSystem
- LaunchDbSystem
- ListDbSystems
- TerminateDbSystem

Database homes:

- GetDbHome
- ListDbHomes

Databases:

- GetDatabase
- ListDatabases

Nodes:

- DbNodeAction: Use this operation to power cycle a node in the DB system.
- ListDbNodes
- GetDbNode

Shapes and database versions:

- ListDbSystemShapes
- ListDbVersions

Configuring a Static Route for Accessing the Object Store

All the traffic in an Exadata DB system is, by default, routed through the data network. To route backup traffic to the backup interface (BONDETH1), you need to configure a static route on *each* of the compute nodes in the cluster. For instructions, see Node Access to Object Storage: Static Route.

Setting Up DNS for a DB System

DNS lets you use hostnames instead of IP addresses to communicate with a DB system. You can use the *Internet and VCN Resolver* (the DNS capability built into the VCN) as described in DNS in Your Virtual Cloud Network. Oracle recommends using a VCN Resolver for DNS name resolution for the client subnet. It automatically resolves the Swift endpoints required for backing up databases, patching, and updating the cloud tooling on an Exadata DB system.

Managing Exadata DB System I/O Resources

This topic explains the I/O Resource Management (IORM) feature and how to enable it, modify the IORM settings, and disable it by using the Console.

About IORM

The I/O Resource Management (IORM) feature allows you to manage how multiple databases share the I/O resources of an Oracle Exadata DB system.

On an Exadata DB system, all databases share dedicated storage servers which include flash storage. By default, the databases are given equal priority with respect to these resources. The Exadata storage management software uses a first come, first served approach for query processing. If a database executes a major query that overloads I/O resources, overall system performance can be slowed down.

IORM allows you to assign priorities to your databases to ensure critical queries are processed first when workloads exceed their resource allocations. You assign priorities by creating directives that specify the number of shares for each database. The number of shares corresponds to a percentage of resources given to that database when I/O resources are stressed.

Directives work together with an overall optimization objective you set for managing the resources. The following objectives are available:

- **Auto** - Recommended. IORM determines the optimization objective and continuously and dynamically determines the optimal settings, based on the workloads observed, and resource plans enabled.
- **Balanced** - For critical OLTP and DSS workloads. This setting balances low disk latency and high throughput. This setting limits disk utilization of large I/Os to a lesser extent than low latency to achieve a balance between good latency and good throughput.
- **High throughput** - For critical DSS workloads that require high throughput.
- **Low latency** - For critical OLTP workloads. This setting provides the lowest possible latency by significantly limiting disk utilization.

For more information about IORM, see Exadata System Software User's Guide.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let database admins manage database systems lets the specified group do everything with databases and related Database resources.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for databases, see Details for the Database Service.

Using the Console

To enable IORM on your Exadata DB system

Enabling IORM includes specifying an optimization objective and configuring your resource

plan directives.

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
3. In the list of DB systems, find the Exadata DB system for which you want to enable IORM, and click its highlighted name.
   The system details are displayed, showing the IORM status as "Disabled."
4. Cick **Enable IORM**.
   It might take a minute for the Enable I/O Resource Management dialog to retrieve the DB system information.
5. Select the objective to apply to the resource plan:
   - **Auto** - (Recommended) Dynamically changes the objective based on the resource plan and observed workloads.
   - **Balanced** - Weighs high throughput and low latency evenly.
   - **High throughput** - Provides the best throughput for DSS workloads.
   - **Low latency** - Provides the best latency for critical OLTP workloads.
6. Configure the resource plan default directive by setting the number of shares. This number of shares is assigned to each database not associated with a specific directive.
7. In the Resource Plan Directives section, add a directive for each database you want to assign a greater or lesser number of shares than the default directive.
   To add a directive, click **+ Additional Directive**, then specify the database and the number of shares for that database.
8. When you are done adding directives, click **Enable**.
   While the IORM configuration settings are being applied, the system details page shows the IORM status as "Updating." The update might take several minutes to complete but should have no impact on your ability to perform normal operations on your DB system. After a successful update, the IORM status shows as "Enabled."

To modify the IORM configuration on your Exadata DB system

Use this procedure to change your IORM settings or to disable IORM.

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
3. In the list of DB systems, find the Exadata DB system for which you want to modify the IORM configuration, and click its highlighted name.
   The system details are displayed, showing the IORM status as "Enabled."
4. Click **Update IORM**.
5. In the Update I/O Resource Management dialog, take one of the following actions:
   - Change your settings - Specify a new objective and adjust your directives, as applicable, and then click **Update**.
   - Disable IORM - Click **Disable IORM**. Disabling IORM removes all your resource plan directives and restores a basic objective for I/O resource management.

   While the new IORM configuration settings are being applied, the system details page shows the IORM status as "Updating." The update might take several minutes to complete but should have no impact on your ability to perform normal operations on your DB system. After a successful update, the IORM status shows as "Enabled" or "Disabled," depending on the action you took.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage the I/O resources of your Exadata DB system.

- ListDbSystems
- GetDbSystem
- GetExadataIormConfig
- UpdateExadataIormConfig

Connecting to an Exadata DB System

This topic explains how to connect to an Exadata DB System using SSH or SQL Developer. How you connect depends on how your cloud network is set up. You can find information on various networking scenarios in Overview of Networking, but for specific recommendations on how you should connect to a database in the cloud, contact your network security administrator.

Prerequisites

For SSH access to a compute node in an Exadata DB System, you'll need the following:

- The full path to the file that contains the private key associated with the public key used when the system was launched.
- The public or private IP address of the DB System. Use the private IP address to connect to the DB system from your on-premises network, or from within the virtual cloud network (VCN). This includes connecting from a host located on-premises connecting through a VPN or FastConnect to your VCN, or from another host in the same VCN. Use the DB System's public IP address to connect to the system from outside the cloud (with no VPN). You can find the IP addresses in the Oracle Cloud Infrastructure Console on the **Database** page.

Connecting to a Compute Node with SSH

You can connect to the compute nodes in an Exadata DB System by using a Secure Shell (SSH) connection. Most UNIX-style systems (including Linux, Solaris, BSD, and OS X) include an SSH client by default. For Windows, you can download a free SSH client called PuTTY from http://www.putty.org.

To connect from a UNIX-style system

Use the following SSH command to access a compute node:

```
$ ssh -i <private key> opc@<DB System IP address>
```

<private key> is the full path and name of the file that contains the private key associated with the Exadata DB System you want to access.

Use the private or public IP address depending on your network configuration. For more information, see Prerequisites.

To connect from a Windows system

1. Open `putty.exe`.
2. In the **Category** pane, select **Session** and enter the following fields:
   - **Host Name (or IP address):** `opc@`*`<ip_address>`*
     Use the compute node's private or public IP address depending on your network configuration. For more information, see Prerequisites.
   - **Connection type:** SSH
   - **Port:** 22
3. In the **Category** pane, expand **Connection**, expand **SSH**, and then click **Auth**, and browse to select your private key.
4. Optionally, return to the **Session** category screen and save this session information for reuse later.
5. Click **Open** to start the session.

To access a database after you connect to the compute node

1. Log in as opc and then sudo to the root user.

```
login as: opc

[opc@ed1db01 ~]$ sudo su -
```

2. Set the environment to the ASM instance. Depending on which node you are connecting to, the ASM instance ID will vary, for example, +ASM1 , +ASM2, and so on.

```
[root@ed1db01 ]# . oraenv
ORACLE_SID = [root] ? +ASM1
The Oracle base has been set to /u01/app/grid
```

3. List the databases on the system.

```
root@ed1db01 ]# srvctl config database -v

cdbm01 /u02/app/oracle/product/12.1.0/dbhome_2 12.1.0.2.0
exadb /u02/app/oracle/product/11.2.0/dbhome_2 11.2.0.4.0
mmdb /u02/app/oracle/product/12.1.0/dbhome_3 12.1.0.2.0
```

4. Connect as the oracle user and get the details about one of the databases using srvctl command.

```
[root@ed1db01 ~]# su - oracle
[oracle@ed1db01 ~]$ . oraenv
ORACLE_SID = [oracle] ? cdbm01
The Oracle base has been set to /u02/app/oracle
[oracle@ed1db01 ~]$ srvctl config database -d cdbm01
Database unique name: cdbm01 <<== DB unique name
Database name:
Oracle home: /u02/app/oracle/product/12.1.0/dbhome_2
Oracle user: oracle
Spfile: +DATAC1/cdbm01/spfilecdbm01.ora
Password file: +DATAC1/cdbm01/PASSWORD/passwd
Domain: data.customer1.oraclevcn.com
Start options: open
Stop options: immediate
Database role: PRIMARY
Management policy: AUTOMATIC
Server pools:
Disk Groups: DATAC1,RECOC1
Mount point paths:
Services:
Type: RAC
Start concurrency:
Stop concurrency:
OSDBA group: dba
OSOPER group: racoper
Database instances: cdbm011,cdbm012 <<== SID
```

```
Configured nodes: ed1db01,ed1db02
Database is administrator managed
```

Connecting to a Database with SQL Developer

You can connect to a database with SQL Developer by using one of the following methods:

- Create a temporary SSH tunnel from your computer to the database. This method provides access only for the duration of the tunnel. (When you are done using the database, be sure to close the SSH tunnel by exiting the SSH session.)
- Open port 1521 for the Oracle default listener by updating the security list used for the DB System. This method provides more durable access to the database. For more information, see Updating the Security List.

After you've created an SSH tunnel or opened port 1521 as described above, you can connect to a Exadata DB System using SCAN IP addresses or public IP addresses, depending on how your network is set up and where you are connecting from. You can find the IP addresses in the Console, in the **Database** details page.

To connect using SCAN IP addresses

You can connect to the database using the SCAN IP addresses if your client is on-premises and you are connecting using a FastConnect or VPN connection. You have the following options:

- Use the private SCAN IP addresses, as shown in the following `tnsnames.ora` example:

```
testdb=
  (DESCRIPTION =
    (ADDRESS_LIST=
      (ADDRESS = (PROTOCOL = TCP)(HOST = <scanIP1>)(PORT = 1521))
      (ADDRESS = (PROTOCOL = TCP)(HOST = <scanIP2>)(PORT = 1521)))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = <dbservice.subnetname.dbvcn.oraclevcn.com>)
    )
  )
```

- Define an external SCAN name in your on-premises DNS server. Your application can resolve this external SCAN name to the DB System's private SCAN IP addresses, and then the application can use a connection string that includes the external SCAN name. In the following `tnsnames.ora` example, `extscanname.example.com` is defined in the on-premises DNS server.

```
testdb =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = <extscanname.example.com>)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = <dbservice.subnetname.dbvcn.oraclevcn.com>)
    )
  )
```

To connect using public IP addresses

You can use the node's public IP address to connect to the database if the client and database are in different VCNs, or if the database is on a VCN that has an internet gateway. However, there are important implications to consider:

- When the client uses the public IP address, the client bypasses the SCAN listener and reaches the node listener, so server side load balancing is not available.
- When the client uses the public IP address, it cannot take advantage of the VIP failover feature. If a node becomes unavailable, new connection attempts to the node will hang until a TCP/IP timeout occurs. You can set client side sqlnet parameters to limit the TCP/IP timeout.

The following `tnsnames.ora` example shows a connection string that includes the CONNECT_TIMEOUT parameter to avoid TCP/IP timeouts.

```
test=
  (DESCRIPTION =
    (CONNECT_TIMEOUT=60)
    (ADDRESS_LIST=
      (ADDRESS = (PROTOCOL = TCP)(HOST = <publicIP1>)(PORT = 1521))
```

```
      (ADDRESS = (PROTOCOL = TCP)(HOST = <publicIP2>)(PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = <dbservice.subnetname.dbvcn.oraclevcn.com>)
    )
  )
```

Updating an Exadata DB System

This topic covers how to update the operating system and the tooling on the database server nodes (also known as "compute nodes") of an Exadata DB system. Review all of the information carefully before you begin the updates.

OS Updates



Important

Starting on March 14, 2019, Exadata DB system images run Oracle Linux 7 (OL7). Previously launched systems are running Oracle Linux 6 (OL6). The underlying infrastructure of existing Exadata DB systems will be patched in the April 2019 time frame. This patch will allow you to upgrade the DB system operating system to OL7.

After you receive notification that your infrastructure patch update is complete, follow the instructions in Patching an Exadata DB System to patch the Oracle Grid Infrastructure and the databases, and then update the OS. Review the minimum software requirements and other details in How to update the Exadata System Software (DomU) to 19c on the Exadata Cloud Service in OCI (Doc ID 2521053.1) before you perform the OS update tasks.

You update the operating systems of Exadata compute nodes by using the `patchmgr` tool. This utility manages the entire update of one or more compute nodes remotely, including running pre-reboot, reboot, and post-reboot steps. You can run the utility from either an Exadata compute node or a non-Exadata server running Oracle Linux. The server on which you run the utility is known as the "driving system." You cannot use the driving system to update itself. Therefore, if the driving system is one of the Exadata compute nodes on a system you are updating, you must run a separate operation on a different driving system to update that server.

The following two scenarios describe typical ways of performing the updates:

Scenario 1: Non-Exadata Driving System

The simplest way to run the update the Exadata system is to use a separate Oracle Linux server to update all Exadata compute nodes in the system.

Scenario 2: Exadata Node Driving System

You can use one Exadata compute node to drive the updates for the rest of the compute nodes in the system, and then use one of the updated nodes to drive the update on the original Exadata driver node.

For example: You are updating a half rack Exadata system, which has four compute nodes - node1, node2, node3, and node4. First, use node1 to drive the updates of node2, node3, and node4. Then, use node2 to drive the update of node1.

The driving system requires root user `SSH` access to each compute node the utility will update.

PREPARING FOR THE OS UPDATES



Warning

Do not install NetworkManager on the DB system. Installing this package and rebooting the system results in severe loss of access to the system.

- Before you begin your updates, review *Exadata Cloud Service Software Versions* (Doc ID 2333222.1) to determine the latest software version and target version to use.
- Some steps in the update process require you to specify a YUM repository. The YUM repository URL is:

  ```
  http://yum-<region_key>.oracle.com/repo/EngineeredSystems/exadata/dbserver/<latest_version>/base/x86_64.
  ```

  Region keys are three-letter abbreviations, for example PHX.
  You can run the following `curl` command to determine the latest version of the YUM repository for your DB system region:

  ```
  curl -s -X GET http://yum-<region_key>.oracle.com/repo/EngineeredSystems/exadata/dbserver/index.html |egrep "18.1."
  ```

This example returns the most current version of the YUM repository for the us-phoenix-1:

```
curl -s -X GET http://yum-phx.oracle.com/repo/EngineeredSystems/exadata/dbserver/index.html
|egrep "18.1."
<a href="18.1.4.0.0/">18.1.4.0.0/</a> 01-Mar-2018 03:36 -
```

- To apply OS updates, the DB system's VCN must be configured to allow access to the YUM repository. For more information, see Security Lists for the Subnets.

To update the OS on all compute nodes of an Exadata DB system

This example procedure assumes the following:

- The system has two compute nodes, `node1` and `node2`.
- The target version is 18.1.4.0.0.180125.3.
- Each of the two nodes is used as the driving system for the update on the other one.

1. Gather the environment details.
   a. `SSH` to `node1` as `root` and run the following command to determine the version of Exadata:

      ```
      [root@node1]# imageinfo -ver
      12.2.1.1.4.171128
      ```

   b. Switch to the grid user, and identify all computes in the cluster.

      ```
      [root@node1]# su - grid
      [grid@node1]$ olsnodes
      node1
      node1
      ```

2. Configure the driving system.
   a. Switch back to the root user on `node1`, check whether a root ssh key pair (`id_rsa` and `id_rsa.pub`) already exists. If not, then generate it.

      ```
      [root@node1 .ssh]#  ls /root/.ssh/id_rsa*
      ls: cannot access /root/.ssh/id_rsa*: No such file or directory
      [root@node1 .ssh]# ssh-keygen -t rsa
      ```

```
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
93:47:b0:83:75:f2:3e:e6:23:b3:0a:06:ed:00:20:a5
root@node1.fraad1client.exadataclientne.oraclevcn.com
The key's randomart image is:
+--[ RSA 2048]----+
|o..     + .      |
|o.     o *       |
|E     . o o      |
| . .     =       |
|  o .   S =      |
|   +     = .     |
|    +   o o      |
|   . .   + .     |
|      ...        |
+-----------------+
```

b. Distribute the public key to the target nodes, and verify this step. In this example, the only target node is `node2`.

```
[root@node1 .ssh]# scp -i ~opc/.ssh/id_rsa ~root/.ssh/id_rsa.pub opc@node2:/tmp/id_
rsa.node1.pub
id_rsa.pub

[root@node2 ~]# ls -al /tmp/id_rsa.node1.pub
-rw-r--r-- 1 opc opc 442 Feb 28 03:33 /tmp/id_rsa.node1.pub
[root@node2 ~]# date
Wed Feb 28 03:33:45 UTC 2018
```

c. On the target node (`node2`, in this example), add the root public key of `node1` to the root `authorized_keys` file.

```
[root@node2 ~]# cat /tmp/id_rsa.node1.pub >> ~root/.ssh/authorized_keys
```

d. Download `dbserver.patch.zip` as `p21634633_12*_Linux-x86-64.zip` onto the driving system (`node1`, in this example), and unzip it. See *dbnodeupdate.sh and*

dbserver.patch.zip: Updating Exadata Database Server Software using the DBNodeUpdate Utility and patchmgr (Doc ID 1553103.1) for information about the files in this .zip.

```
[root@node1 patch]# mkdir /root/patch
[root@node1 patch]# cd /root/patch
[root@node1 patch]# unzip p21634633_181400_Linux-x86-64.zip
Archive:  p21634633_181400_Linux-x86-64.zip   creating: dbserver_patch_5.180228.2/
   creating: dbserver_patch_5.180228.2/ibdiagtools/
  inflating: dbserver_patch_5.180228.2/ibdiagtools/cable_check.pl
  inflating: dbserver_patch_5.180228.2/ibdiagtools/setup-ssh
  inflating: dbserver_patch_5.180228.2/ibdiagtools/VERSION_FILE
 extracting: dbserver_patch_5.180228.2/ibdiagtools/xmonib.sh
  inflating: dbserver_patch_5.180228.2/ibdiagtools/monitord
  inflating: dbserver_patch_5.180228.2/ibdiagtools/checkbadlinks.pl
   creating: dbserver_patch_5.180228.2/ibdiagtools/topologies/
  inflating: dbserver_patch_5.180228.2/ibdiagtools/topologies/VerifyTopologyUtility.pm
  inflating: dbserver_patch_5.180228.2/ibdiagtools/topologies/verifylib.pm
  inflating: dbserver_patch_5.180228.2/ibdiagtools/topologies/Node.pm
  inflating: dbserver_patch_5.180228.2/ibdiagtools/topologies/Rack.pm
  inflating: dbserver_patch_5.180228.2/ibdiagtools/topologies/Group.pm
  inflating: dbserver_patch_5.180228.2/ibdiagtools/topologies/Switch.pm
  inflating: dbserver_patch_5.180228.2/ibdiagtools/topology-zfs
  inflating: dbserver_patch_5.180228.2/ibdiagtools/dcli
   creating: dbserver_patch_5.180228.2/ibdiagtools/netcheck/
  inflating: dbserver_patch_5.180228.2/ibdiagtools/netcheck/remoteScriptGenerator.pm
  inflating: dbserver_patch_5.180228.2/ibdiagtools/netcheck/CommonUtils.pm
  inflating: dbserver_patch_5.180228.2/ibdiagtools/netcheck/SolarisAdapter.pm
  inflating: dbserver_patch_5.180228.2/ibdiagtools/netcheck/LinuxAdapter.pm
  inflating: dbserver_patch_5.180228.2/ibdiagtools/netcheck/remoteLauncher.pm
  inflating: dbserver_patch_5.180228.2/ibdiagtools/netcheck/remoteConfig.pm
  inflating: dbserver_patch_5.180228.2/ibdiagtools/netcheck/spawnProc.pm
  inflating: dbserver_patch_5.180228.2/ibdiagtools/netcheck/runDiagnostics.pm
  inflating: dbserver_patch_5.180228.2/ibdiagtools/netcheck/OSAdapter.pm
  inflating: dbserver_patch_5.180228.2/ibdiagtools/SampleOutputs.txt
  inflating: dbserver_patch_5.180228.2/ibdiagtools/infinicheck
  inflating: dbserver_patch_5.180228.2/ibdiagtools/ibping_test
  inflating: dbserver_patch_5.180228.2/ibdiagtools/tar_ibdiagtools
  inflating: dbserver_patch_5.180228.2/ibdiagtools/verify-topology
  inflating: dbserver_patch_5.180228.2/installfw_exadata_ssh
```

```
   creating: dbserver_patch_5.180228.2/linux.db.rpms/
  inflating: dbserver_patch_5.180228.2/md5sum_files.lst
  inflating: dbserver_patch_5.180228.2/patchmgr
  inflating: dbserver_patch_5.180228.2/xcp
  inflating: dbserver_patch_5.180228.2/ExadataSendNotification.pm
  inflating: dbserver_patch_5.180228.2/ExadataImageNotification.pl
  inflating: dbserver_patch_5.180228.2/kernelupgrade_oldbios.sh
  inflating: dbserver_patch_5.180228.2/cellboot_usb_pci_path
  inflating: dbserver_patch_5.180228.2/exadata.img.env
  inflating: dbserver_patch_5.180228.2/README.txt
  inflating: dbserver_patch_5.180228.2/exadataLogger.pm
  inflating: dbserver_patch_5.180228.2/patch_bug_26678971
  inflating: dbserver_patch_5.180228.2/dcli
  inflating: dbserver_patch_5.180228.2/patchReport.py
 extracting: dbserver_patch_5.180228.2/dbnodeupdate.zip
   creating: dbserver_patch_5.180228.2/plugins/
  inflating: dbserver_patch_5.180228.2/plugins/010-check_17854520.sh
  inflating: dbserver_patch_5.180228.2/plugins/020-check_22468216.sh
  inflating: dbserver_patch_5.180228.2/plugins/040-check_22896791.sh
  inflating: dbserver_patch_5.180228.2/plugins/000-check_dummy_bash
  inflating: dbserver_patch_5.180228.2/plugins/050-check_22651315.sh
  inflating: dbserver_patch_5.180228.2/plugins/005-check_22909764.sh
  inflating: dbserver_patch_5.180228.2/plugins/000-check_dummy_perl
  inflating: dbserver_patch_5.180228.2/plugins/030-check_24625612.sh
  inflating: dbserver_patch_5.180228.2/patchmgr_functions
  inflating: dbserver_patch_5.180228.2/exadata.img.hw
  inflating: dbserver_patch_5.180228.2/libxcp.so.1
  inflating: dbserver_patch_5.180228.2/imageLogger
  inflating: dbserver_patch_5.180228.2/ExaXMLNode.pm
  inflating: dbserver_patch_5.180228.2/fwverify
```

e. Create the `dbs_group` file that contains the list of compute nodes to update. Include the nodes listed after running the `olsnodes` command in step 1 except for the driving system node. In this example, `dbs_group` should include only `node2`.

```
[root@node1 patch]# cd /root/patch/dbserver_patch_5.180228
[root@node1 dbserver_patch_5.180228]# cat dbs_group
node2
```

3. Run a patching precheck operation.

```
patchmgr -dbnodes dbs_group -precheck -yum_repo <yum_repository> -target_version <target_version>
-nomodify_at_prereq
```



Important

You must run the precheck operation with the `-nomodify_at_prereq` option to prevent any changes to the system that could impact the backup you take in the next step. Otherwise, the backup might not be able to roll back the system to its original state, should that be necessary.

The output should look like the following example:

```
[root@node1 dbserver_patch_5.180228]# ./patchmgr -dbnodes dbs_group -precheck -yum_repo
http://yum-phx.oracle.com/repo/EngineeredSystems/exadata/dbserver/18.1.4.0.0/base/x86_64 -target_
version 18.1.4.0.0.180125.3  -nomodify_at_prereq



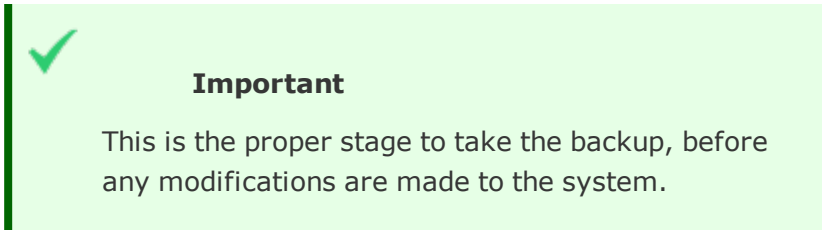
********************************************************************************************
***********
NOTE    patchmgr release: 5.180228 (always check MOS 1553103.1 for the latest release of
dbserver.patch.zip)
NOTE
WARNING Do not interrupt the patchmgr session.
WARNING Do not resize the screen. It may disturb the screen layout.
WARNING Do not reboot database nodes during update or rollback.
WARNING Do not open logfiles in write mode and do not try to alter them.

********************************************************************************************
***********
2018-02-28 21:22:45 +0000        :Working: DO: Initiate precheck on 1 node(s)
2018-02-28 21:24:57 +0000        :Working: DO: Check free space and verify SSH equivalence for
the root user to node2
2018-02-28 21:26:15 +0000        :SUCCESS: DONE: Check free space and verify SSH equivalence for
the root user to node2
```

```
2018-02-28 21:26:47 +0000        :Working: DO: dbnodeupdate.sh running a precheck on node(s).
2018-02-28 21:28:23 +0000        :SUCCESS: DONE: Initiate precheck on node(s).
```

4. Back up the current system.

```
patchmgr -dbnodes dbs_group -backup -yum_repo <yum_repository> -target_version <target_version>
-allow_active_network_mounts
```



The output should look like the following example:

```
[root@node1 dbserver_patch_5.180228]#  ./patchmgr -dbnodes dbs_group -backup  -yum_repo
http://yum-phx.oracle.com/repo/EngineeredSystems/exadata/dbserver/18.1.4.0.0/base/x86_64 -target_
version 18.1.4.0.0.180125.3 -allow_active_network_mounts


********************************************************************************************************
***********
NOTE    patchmgr release: 5.180228 (always check MOS 1553103.1 for the latest release of
dbserver.patch.zip)
NOTE
WARNING Do not interrupt the patchmgr session.
WARNING Do not resize the screen. It may disturb the screen layout.
WARNING Do not reboot database nodes during update or rollback.
WARNING Do not open logfiles in write mode and do not try to alter them.

********************************************************************************************************
***********
2018-02-28 21:29:00 +0000        :Working: DO: Initiate backup on 1 node(s).
2018-02-28 21:29:00 +0000        :Working: DO: Initiate backup on node(s)
2018-02-28 21:29:01 +0000        :Working: DO: Check free space and verify SSH equivalence for
the root user to node2
2018-02-28 21:30:18 +0000        :SUCCESS: DONE: Check free space and verify SSH equivalence for
the root user to node2
2018-02-28 21:30:51 +0000        :Working: DO: dbnodeupdate.sh running a backup on node(s).
```

```
2018-02-28 21:35:50 +0000        :SUCCESS: DONE: Initiate backup on node(s).
2018-02-28 21:35:50 +0000        :SUCCESS: DONE: Initiate backup on 1 node(s).
```

5. Remove all custom RPMs from the target compute nodes that will be updated. Custom RPMs are reported in precheck results. They include RPMs that were manually installed after the system was provisioned.



Note

- If you are updating the system from version 12.1.2.3.4.170111, and the precheck results include `krb5-workstation-1.10.3-57.el6.x86_64`, remove it. (This item is considered a custom RPM for this version.)
- Do **not** remove `exadata-sun-vm-computenode-exact` or `oracle-ofed-release-guest`. These two RPMs are handled automatically during the update process.

6. Run the `nohup` command to perform the update.

```
nohup patchmgr -dbnodes dbs_group -upgrade -nobackup -yum_repo <yum_repository> -target_version
<target_version> -allow_active_network_mounts &
```

The output should look like the following example:

```
[root@node1 dbserver_patch_5.180228]# nohup ./patchmgr -dbnodes dbs_group -upgrade -nobackup  -
yum_repo  http://yum-phx.oracle.com/repo/EngineeredSystems/exadata/dbserver/18.1.4.0.0/base/x86_
64 -target_version 18.1.4.0.0.180125.3  -allow_active_network_mounts &


********************************************************************************************
***********
NOTE    patchmgr release: 5.180228 (always check MOS 1553103.1 for the latest release of
dbserver.patch.zip)
NOTE
NOTE    Database nodes will reboot during the update process.
```

```
NOTE
WARNING Do not interrupt the patchmgr session.
WARNING Do not resize the screen. It may disturb the screen layout.
WARNING Do not reboot database nodes during update or rollback.
WARNING Do not open logfiles in write mode and do not try to alter them.

********************************************************************************************
********

2018-02-28 21:36:26 +0000        :Working: DO: Initiate prepare steps on node(s).
2018-02-28 21:36:26 +0000        :Working: DO: Check free space and verify SSH equivalence for
the root user to node2
2018-02-28 21:37:44 +0000        :SUCCESS: DONE: Check free space and verify SSH equivalence for
the root user to node2
2018-02-28 21:38:43 +0000        :SUCCESS: DONE: Initiate prepare steps on node(s).
2018-02-28 21:38:43 +0000        :Working: DO: Initiate update on 1 node(s).
2018-02-28 21:38:43 +0000        :Working: DO: Initiate update on node(s)
2018-02-28 21:38:49 +0000        :Working: DO: Get information about any required OS upgrades
from node(s).
2018-02-28 21:38:59 +0000        :SUCCESS: DONE: Get information about any required OS upgrades
from node(s).
2018-02-28 21:38:59 +0000        :Working: DO: dbnodeupdate.sh running an update step on all
nodes.
2018-02-28 21:48:41 +0000        :INFO   : node2 is ready to reboot.
2018-02-28 21:48:41 +0000        :SUCCESS: DONE: dbnodeupdate.sh running an update step on all
nodes.
2018-02-28 21:48:41 +0000        :Working: DO: Initiate reboot on node(s)
2018-02-28 21:48:57 +0000        :SUCCESS: DONE: Initiate reboot on node(s)
2018-02-28 21:48:57 +0000        :Working: DO: Waiting to ensure node2 is down before reboot.
2018-02-28 21:56:18 +0000        :Working: DO: Initiate prepare steps on node(s).
2018-02-28 21:56:19 +0000        :Working: DO: Check free space and verify SSH equivalence for
the root user to node2
2018-02-28 21:57:37 +0000        :SUCCESS: DONE: Check free space and verify SSH equivalence for
the root user to node2
2018-02-28 21:57:42 +0000        :SEEMS ALREADY UP TO DATE: node2
2018-02-28 21:57:43 +0000        :SUCCESS: DONE: Initiate update on node(s)
```

7. After the update operation completes, verify the version of the kernel on the compute node that was updated.

```
[root@node2 ~]# imageinfo -ver
18.1.4.0.0.180125.3
```

8. If the driving system is a compute node that needs to be updated (as in this example), repeat steps 2 through 7 of this procedure using an updated compute node as the driving system to update the remaining compute node. In this example update, you would use `node2` to update `node1`.
9. On each compute node, run the `uptrack-install` command as root to install the available ksplice updates.

```
uptrack-install --all -y
```

Updating Tooling on an Exadata DB System

You can update the cloud-specific tooling included on an Exadata DB system compute node by downloading and applying an RPM file containing the latest version of the tools.



Note

Oracle highly recommends that you maintain the same version of cloud tooling across your Exadata DB system environment. Perform the following procedure on every compute node in the Exadata DB system.

PREREQUISITE

The compute nodes in the Exadata DB system must be configured to access the Oracle Cloud Infrastructure Object Storage service. For more information, see Node Access to Object Storage: Static Route.

UPDATING THE CLOUD TOOLING ON EACH COMPUTE NODE

The method for updating the tooling depends on the tooling release that is currently installed on the compute node. Regardless of the method you use, be sure to repeat the update process on each compute node in the cluster.

To check the installed tooling release

1. Connect to the compute node as the opc user.
2. Start a root-user command shell.

```
$ sudo -s
#
```

3. Use the following command to display information about the installed cloud tooling and note the release label, shown in red in the example that follows.

```
# rpm -qa|grep -i dbaastools_exa

dbaastools_exa-1.0-1+18.1.2.1.0_180511.0801.x86_64
```

In this example, the release version is **18.1.2.1.0_180511.0801**.

To update the tooling if the release label is higher than 17430

1. Check whether any cloud tooling updates are available.

```
# /var/opt/oracle/exapatch/exadbcpatchsm -list_tools
                                    :
          'current_version' => '18.1.2.1.180511',
          'last_async_precheck_patch_id' => ' ',
          'current_patch' => '180511',
          'last_async_apply_patch_id' => ' ',
          'patches' => [
                                     :

                            'patchid' => '18.1.4.1.0_180716.0000',
                            'last_precheck_txnid' => '',
                            'description' => 'DBAAS Tools for Oracle Public Cloud'
                          },
                          {
                            'patchid' => '18.1.4.1.0_180718.0000',
                            'last_precheck_txnid' => '',
                            'description' => 'DBAAS Tools for Oracle Public Cloud'
                          }
```

```
                    ]
        };
```

2. Examine the command response and determine the patch ID of the available cloud tooling update.

   The patch ID is listed in the patches group as the `patchid` value.

   Cloud tooling updates are cumulative. So if multiple updates are available (as in the example in step 1), you can simply install the latest update. There is no need to install all of the updates in order.

3. If an available patch is newer than the currently installed tools, download and apply the patch containing the cloud tooling update.

   ```
   # /var/opt/oracle/exapatch/exadbcpatchsm -toolsinst_async <patchid>
   ```

   where `patchid` is the patch ID that you located in the previous step.

   For example, the following command applies the latest patch available from the results in step 1:

   ```
   # /var/opt/oracle/exapatch/exadbcpatchsm -toolsinst_async 18.1.4.1.0_180718.0000
   ```

4. Repeat the previous steps on each compute node in the Exadata DB system.



Note

The `exadbcpatchsm` utility runs as a background process which outputs a transaction ID and immediately returns control to the user. Command output is written to a log file. You can monitor the progress of operations by executing:

```
# /var/opt/oracle/exapatch/exadbcpatchsm -get_status
transactionid
```

To update the tooling if the release label is 17430 or lower

1. Download the RPM file using the Swift object storage API endpoint URL for your region.

   ```
   wget <swift_API_endpoint>/v1/exadata/patches/dbaas_patch/shome/dbaastools_exa.rpm
   ```

   The following example downloads the RPM file from the us-phoenix-1.

   ```
   wget https://swiftobjectstorage.us-phoenix-1.oraclecloud.com/v1/exadata/patches/dbaas_
   patch/shome/dbaastools_exa.rpm
   ```

   See API Reference and Endpoints for the Swift API endpoint for your region.

2. Apply the RPM file.

   ```
   # rpm -ev dbaastools_exa
   # rpm -ivh dbaastools_exa.rpm
   ```

3. Repeat the previous steps on each compute node in the Exadata DB system.

Patching an Exadata DB System

This topic explains how to use the `exadbcpatchmulti` utility to perform patching operations for Oracle Grid Infrastructure and Oracle Database on an Exadata DB system. The `exadbcpatchmulti` utility is located in `/var/opt/oracle/exapatch` on every compute node. The utility requires root administration privileges.



Note

You must update the cloud specific tooling on all the compute nodes in your Exadata DB system before performing the following procedures. For more information, see Updating an Exadata DB System.

Prerequisites

Patches are stored in Oracle Cloud Infrastructure Object Storage, so the Exadata DB system

requires access to that service. To enable this access, Oracle recommends using a service gateway with the VCN. For more information, see Network Setup for Exadata DB Systems. In that topic, pay particular attention to:

- Service Gateway for the VCN
- Node Access to Object Storage: Static Route
- Rule for Backup Subnet's Custom Security List

Managing Patches

To list available patches

You can produce a list of available patches using the `exadbcpatchmulti` command:

1. Connect to the compute node as the `opc` user.
   For detailed instructions, see Connecting to an Exadata DB System.
2. Start a root-user command shell:

   ```
   $ sudo -s
   #
   ```

3. Execute the `exadbcpatchmulti` command with the `-list_patches` action:

   ```
   # /var/opt/oracle/exapatch/exadbcpatchmulti -list_patches
   -sshkey=sshkey_file -oh=hostname:oracle_home
   ```

   where:

   - `-sshkey` specifies the location of the SSH private key of the `opc` user, which is used to connect to compute nodes in the cluster. This is an optional parameter.
     `-oh` specifies a compute node and Oracle home directory for which you want to list the available patches. In this context, an Oracle home directory may be an Oracle Database home directory or the Oracle Grid Infrastructure home directory.

   For example:

   ```
   # /var/opt/oracle/exapatch/exadbcpatchmulti -list_patches -oh=exaverify-
   73z1v1:/u02/app/oracle/product/12.1.0/dbhome_2 -sshkey=/root/y.priv
   INFO: non async case
   ```

```
INFO: cmd is:  /var/opt/oracle/exapatch/exadbcpatchsm  -list_patches  -
oh=/u02/app/oracle/product/12.1.0/dbhome_2
INFO: non async case
INFO: cmd is: /var/opt/oracle/exapatch/exadbcpatch -list_patches -patch_
homes=/u02/app/oracle/product/12.1.0/dbhome_2
Starting EXADBCPATCH
Logfile is /var/opt/oracle/log/exadbcpatch/exadbcpatch_2017-07-03_19:40:49.log
Config file is /var/opt/oracle/exapatch/exadbcpatch.cfg
     INFO: dbversion detected : 12102
     INFO: patching type : psu
     INFO: using default values for oci_user
     INFO: using default value for oci_passwd
     INFO: images available for patching
     12.1.0.2.170117, ee
     $VAR1 = {
          'last_async_precheck_txn_id' => ' ',
          'last_async_apply_txn_id' => ' ',
          'errmsg' => 'no applicable patches found',
          'err' => '-1',
          'current_version' => '12.1.0.2.170117',
          'last_async_precheck_patch_id' => '',
          'current_patch' => '24968615',
          'last_async_apply_patch_id' => '',
          'patches' => []
          };
<json begin>{"last_async_precheck_txn_id":" ","last_async_apply_txn_id":" ","err":"-
1","errmsg":"no applicable patches found","current_version":"12.1.0.2.170117","last_async_
precheck_patch_id":"","current_patch":"24968615","last_async_apply_patch_id":"","patches":
[]}<json end>
```



Note

The list of available patches is determined by interrogating the database to establish the patches that have already been applied. When a patch is applied, the corresponding database entry is made as part of the SQL patching operation, which is executed at the end of the patch workflow. Therefore, the list of available patches may include partially applied patches along with patches that are currently being applied.

4. Exit the root-user command shell.

```
# exit
$
```

To check prerequisites before applying a patch

You can perform the prerequisites-checking operation using the `exadbcpatchmulti` command as follows:

1. Connect to the compute node as the `opc` user.
2. Start a root-user command shell:

```
$ sudo -s
#
```

3. Execute the `exadbcpatchmulti` command with the `-precheck_async` action:

```
# /var/opt/oracle/exapatch/exadbcpatchmulti -precheck_async patchid
-sshkey=sshkey_file
-instance1=hostname:oracle_home1[,oracle_home2 ...]
[-instance2=hostname:oracle_home1[,oracle_home2 ...] ...]
```

where:

- `patchid` identifies the patch to be pre-checked.
- `-sshkey` specifies the location of the SSH private key of the `opc` user, which is used to connect to compute nodes in the cluster.
- `-instanceN` specifies a compute node and one or more Oracle home directories that are subject to the patching operation. In this context, an Oracle home directory may be an Oracle Database home directory or the Oracle Grid Infrastructure home directory.

For example:

```
# /var/opt/oracle/exapatch/exadbcpatchmulti -precheck_async 12345678
-sshkey=/home/opc/.ssh/id_rsa
-instance1=hostname1:/u01/app/12.1.0.2/grid,/u01/app/oracle/product/12.1.0.2/dbhome_1
```

4. Exit the root-user command shell:

```
# exit
$
```

To apply a patch

You can apply a patch by using the `exadbcpatchmulti` command.

The patching operation:

- Can be used to patch some or all of your compute nodes using one command.
- Coordinates multi-node patching in a rolling manner.
- Can execute patch-related SQL after patching all the compute nodes in the cluster.

You can perform a patching operation using the `exadbcpatchmulti` command as follows:

1. Connect to the compute node as the `opc` user.
2. Start a root-user command shell:

```
$ sudo -s
#
```

3. Execute the `exadbcpatchmulti` command with the `-apply_async` action:

```
# /var/opt/oracle/exapatch/exadbcpatchmulti -apply_async patchid
-sshkey=sshkey_file
-instance1=hostname:oracle_home1 [, oracle_home2 ...]
[-instance2=hostname:oracle_home1 [,oracle_home2 ...] ...]
[-run_datasql=1]
```

where:

- `patchid` identifies the patch to be applied.
- `-sshkey` specifies the location of the SSH private key of the `opc` user, which is used to connect to compute nodes in the cluster.
- `-instance`*N* specifies a compute node and one or more Oracle home directories that are subject to the patching operation. In this context, an Oracle home directory may be an Oracle Database home directory or the Oracle Grid Infrastructure home directory.
- `-run_datasql=1` instructs the `exadbcpatchmulti` command to execute patch-related SQL commands.



Note

- Patch-related SQL should only be executed after all of the compute nodes are patched. Therefore, take care not to specify this argument if you are patching a subset of nodes and further nodes remain to be patched.
- This argument can only be specified in conjunction with a patching operation on a set of compute nodes. Therefore, if you have patched all of your nodes and you did not specify this argument, you will need to manually execute the SQL commands associated with the patch. Refer to the patch documentation for further details.

For example:

```
# /var/opt/oracle/exapatch/exadbcpatchmulti -apply_async 23456789
-sshkey=/home/opc/.ssh/id_rsa
-instance1=hostname1:/u01/app/oracle/product/12.1.0.2/dbhome_1
-instance2=hostname2:/u01/app/oracle/product/12.1.0.2/dbhome_1
-run_datasql=1
```

4. Exit the root-user command shell:

```
# exit
$
```

To list applied patches

You can produce a list of applied patches to determine which patches have been applied.

You can use the `opatch` utility to determine the patches that have been applied to an Oracle Database or Grid Infrastructure installation.

To produce a list of applied patches for an Oracle Database installation:

1. Connect to a compute node as the `oracle` user.
2. Set the ORACLE_HOME variable to the location of the Oracle Database installation you wish to examine. For example:

```
$ export ORACLE_HOME=/u01/app/oracle/product/12.1.0.2/dbhome_1
```

3. Execute the `opatch` command with the `lspatches` option:

```
$ $ORACLE_HOME/OPatch/opatch lspatches
```

To produce a list of applied patches for Oracle Grid Infrastructure:

1. Connect to a compute node as the `opc` user.
2. Become the `grid` user:

```
$ sudo -s
# su - grid
```

3. Execute the `opatch` command with the `lspatches` option:

```
$ $ORACLE_HOME/OPatch/opatch lspatches
```

To roll back a patch

You can roll back a patch or failed patch attempt on a by using the `exadbcpatchmulti` command.

The patch rollback operation:

- Can be used to roll back a patch on some or all of your compute nodes using one command.
- Coordinates multi-node operations in a rolling manner.

- Can execute rollback-related SQL after rolling back the patch on all the compute nodes in the cluster.

You can perform a patch rollback operation using the `exadbcpatchmulti` command as follows:

1. Connect to the compute node as the `opc` user.
2. Start a root-user command shell:

   ```
   $ sudo -s
   #
   ```

3. Execute the `exadbcpatchmulti` command with the `-rollback_async` action:

   ```
   # /var/opt/oracle/exapatch/exadbcpatchmulti -rollback_async patchid
   -sshkey=sshkey_file
   -instance1=hostname:oracle_home1[,oracle_home2 ...]
   [-instance2=hostname:oracle_home1[,oracle_home2 ...] ...]
   [-run_datasql=1]
   ```

   where:

   - `patchid` identifies the patch to be rolled back.
   - `-sshkey` specifies the location of the SSH private key of the `opc` user, which is used to connect to compute nodes in the cluster.
   - `-instanceN` specifies a compute node and one or more Oracle home directories that are subject to the rollback operation. In this context, an Oracle home directory may be an Oracle Database home directory or the Oracle Grid Infrastructure home directory.
   - `-run_datasql=1` instructs the `exadbcpatchmulti` command to execute rollback-related SQL commands.



Note

- Rollback-related SQL should only be executed after all of the compute nodes are rolled back. Therefore, take care not to specify this argument if you are rolling back a subset of nodes and further nodes remain to be rolled back.
- This argument can only be specified in conjunction with a rollback operation on a set of compute nodes. Therefore, if you have rolled back all of your nodes and you did not specify this argument, you will need to manually execute the SQL commands associated with the rollback operation. Refer to the patch documentation for further details.

For example:

```
# /var/opt/oracle/exapatch/exadbcpatchmulti -rollback_async 34567890
-sshkey=/home/opc/.ssh/id_rsa
-instance1=hostname1:/u01/app/12.1.0.2/grid
-instance2=hostname2:/u01/app/12.1.0.2/grid
-run_datasql=1
```

4. Exit the root-user command shell:

```
# exit
$
```

Monitoring a Database on an Exadata DB System

This topic explains how to access Enterprise Manager Database Express and Enterprise Manager Database Control, which are web-based tools for managing Oracle Database.

Accessing Enterprise Manager Database Express 12c

Enterprise Manager Database Express 12c (EM Express) is available on Exadata DB system database deployments created using Oracle Database 12c Release 1 (12.1) or later.

How you access EM Express depends on whether you want to manage a CDB or PDB.

- **To manage the CDB.** When a database deployment is created, Database automatically sets port 5500 on the deployment's compute nodes for EM Express access to the CDB.
- **To manage a PDB.** For an Oracle Database 12.2 or later deployment, a single port (known as the global port) is automatically set on the deployment's compute nodes. The global port lets you use EM Express to connect to all of the PDBs in the CDB using the HTTPS port for the CDB.
  For an Oracle Database 12.1 deployment, you must manually set a port on the deployment's compute nodes for each PDB you want to manage using EM Express.

For both CDBs and PDBs, you must add the port to a security list as described in Updating the Security List.

To confirm the port that is in use for a specific database, connect to the database as a database administrator and execute the query shown in the following example:

```
SQL> select dbms_xdb_config.getHttpsPort() from dual;


DBMS_XDB_CONFIG.GETHTTPSPORT()
------------------------------
                          5502
```

Setting the Port for EM Express to Manage a PDB (Oracle Database 12.1 Only)

In Oracle Database 12c Release 1, a unique HTTPS port must be configured for the root container (CDB) and each PDB that you manage using EM Express.

To configure a HTTPS port so that you can manage a PDB with EM Express:

1. Invoke SQL*Plus and log in to the PDB as the SYS user with SYSDBA privileges.
2. Execute the `DBMS_XDB_CONFIG.SETHTTPSPORT` procedure.

```
SQL> exec dbms_xdb_config.sethttpsport(port-number)
```

ACCESSING EM EXPRESS

Before you access EM Express, add the port to the security list. See Updating the Security List.

After you update the security list, you can access EM Express by directing your browser to the URL `https://<node-ip-address>:<port>/em`, where `node-ip-address` is the public IP address of the compute node hosting EM Express, and `port` is the EM Express port used by the database.

Accessing Enterprise Manager 11g Database Control

Enterprise Manager 11g Database Control (Database Control) is available on Exadata DB system database deployments created using Oracle Database 11g Release 2. Database Control is allocated a unique port number for each database deployment. By default, access to Database Control is provided using port 1158 for the first deployment. Subsequent deployments are allocated ports in a range starting with 5500, 5501, 5502, and so on.

You can confirm the Database Control port for a database by searching for `REPOSITORY_URL` in the `$ORACLE_HOME/host_sid/sysman/config/emd.properties` file.

Before you access Database Control, add the port for the database to the security list associated with the Exadata DB system's client subnet. For more information, see Updating the Security List.

After you update the security list, you can access Database Control by directing your browser to the URL `https://<node-ip-address>:<port>/em`, where `node-ip-address` is the public IP address of the compute node hosting Database Control, and `port` is the Database Control port used by the database.

Updating the Security List

Before you can access EM Express or Database Control, you must add the port for the database to the security list associated with the Exadata DB system's data (client) subnet. To update an existing security list, complete the following steps using the Console:

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. Locate the DB system in the list.
4. Note the DB system's **Client Subnet** name and click its **Virtual Cloud Network**.
5. Locate the subnet in the list, and then click its security list under **Security Lists**.
6. Click **Edit All Rules** and add an ingress rule with source type=CIDR, source CIDR=*<source CIDR>*, protocol=TCP, and port=*<port number or port range>*.
   The source CIDR should be the CIDR block that includes the ports you open for the client connection.

For detailed information about creating or updating a security list, see Security Lists.

Managing Exadata Databases

When you launch an Exadata DB system, an initial database is created in that system. You can create additional databases in that DB system at any time by using the Console or the Oracle Cloud Infrastructure API.

When you add a database to an Exadata DB system, the database versions you can select from depend on the current patch level of that DB system. You might have to patch your DB system to add later database versions. For information about patching the DB system, see Patching an Exadata DB System.

Each new database is created in a separate database home.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

You can also add and remove databases, and perform other management tasks on a database by using command line utilities. For information and instructions on how to use these utilities, see Managing Exadata Databases Manually.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let database admins manage database systems lets the specified group do everything with databases and related Database resources.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for databases, see Details for the Database Service.



Note

See Known Issues for information about using the Oracle Cloud Infrastructure Console, API, or CLI to manage Exadata DB systems if your system was provisioned before June 15, 2018.

Using the Console

To create a new database in an existing Exadata DB system



Note

If IORM is enabled on the DB system, the default



directive will apply to the new database and system performance might be impacted. Oracle recommends that you review the IORM settings and make applicable adjustments to the configuration after the new database is provisioned.

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. In the list of DB systems, find the Exadata DB system in which you want to create the database, and then click its name to display details about it.
4. Click **Create Database**.
5. In the **Create Database** dialog, enter the following:
   - **Database Name:** The name for the database. The database name must begin with an alphabetic character and can contain a maximum of eight alphanumeric characters. Special characters are not permitted.
   - **Database Version:** The version of the database. You can mix database versions on the Exadata DB system.
   - **PDB Name** (Optional) For version 12.1.0.2 and later, you can specify the name of the pluggable database. The PDB name must begin with an alphabetic character, and can contain a maximum of 8 alphanumeric characters. The only special character permitted is the underscore (_).
   - **Admin Password:** A strong password for SYS, SYSTEM, TDE wallet, and PDB Admin. The password must be nine to thirty characters and contain at least two uppercase, two lowercase, two numeric, and two special characters. The special characters must be _, #, or -. In addition, this password must not contain the name of the tenancy or any reserved words, such as "Oracle" or "Table," regardless of casing.
   - **Confirm Admin Password:** Re-enter the database admin password.

- **Database Workload:** Select the workload type that best suits your application.
  - **Online Transactional Processing (OLTP)** configures the database for a transactional workload, with a bias towards high volumes of random data access.
  - **Decision Support System (DSS)** configures the database for a decision support or data warehouse workload, with a bias towards large data scanning operations.

6. Optionally, you can specify the character set and/or national character set for this database. Click **Show Advanced Options** to set these parameters. The defaults are AL32UTF8 and AL16UTF16, respectively.
7. Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
8. Click **Create Database**.

When the database creation is complete, the status changes from Provisioning to Available.

To terminate a database

Oracle recommends that you create a final backup before you terminate any production (non-test) database. See Backing Up an Exadata Database to learn how to back up an Exadata database.

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. In the list of DB Systems, find the DB System that contains the database you want to terminate, and then click its name to display details about it.
4. In the list of databases, find the database, click the Actions icon (three dots) and then

click **Terminate**.

5. Type the name of the database to confirm the termination.
6. Click **Terminate Database**.
   The database's status indicates Terminating.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage databases.

Database homes:

- ListDbHomes
- GetDbHome
- CreateDbHome
- DeleteDbHome

Databases:

- ListDatabases
- GetDatabase

For the complete list of APIs for the Database service, see Database Service API.

Managing Exadata Databases Manually

Exadata DB systems include these command line tools for performing various tasks to manage individual databases:

- `dbaasapi` - For adding and removing databases from the Exadata DB system. See Using dbaasapi.

- `dbaascli` - For a variety of life-cycle and administration operations such as:
    - Starting and stoping a database
    - Starting and stoping the Oracle Net listener
    - Viewing information about Oracle Homes
    - Moving a database to another Oracle Home
    - Deleting an unused Oracle Home
    - Performing database configuration changes
    - Managing Oracle Database software images
    - Managing pluggable databases (PDBs)
    - Performing database recovery
    - Rotating the master encryption key

  For details about how to use this CLI, see The dbaascli Utility.

Using dbaasapi

You can use the `dbaasapi` command line utility to create and delete databases on an Exadata DB system. The utility operates like a REST API. It reads a JSON request body and produces a JSON response body in an output file.

The utility is located in the `/var/opt/oracle/dbaasapi/` directory on the compute nodes and must be run as the root user.

To learn how to add or remove Exadata databases by using the Oracle Cloud Infrastructure Console or API instead, see Managing Exadata Databases.



Note

You must update the cloud-specific tooling on all the compute nodes in your Exadata DB system before performing the following procedures. For more information, see Updating an Exadata DB System.

Only one `dbaasapi` operation can execute at a given time. Oracle recommends that you check the status of an operation to ensure it completed before you run another operation.

Prerequisites

If you plan to create a database and store its backups in the Oracle Cloud Infrastructure Object Storage, refer to the prerequisites in Backing Up an Exadata Database, and ensure that the system meets the networking requirements for backing up to Object Storage. Review the Create Database Parameters and gather the information you'll need to supply in the input file you create for the `dbaasapi` operation.



Warning

Oracle recommends that you avoid specifying parameter values that include confidential information when you use the `dbaasapi` commands.

Creating a Database

The following procedure creates directory called `dbinput`, a sample input file called `myinput.json`, and a sample output file called `createdb.out`.

1. SSH to a compute node in the Exadata DB system.

```
ssh -i <private_key_path> opc@<node_ip_address>
```

2. Log in as opc and then sudo to the root user.

```
login as: opc

[opc@dbsys ~]$ sudo su -
```

3. Make a directory for the input file and change to the directory.

```
[root@dbsys ~]# mkdir -p /home/oracle/dbinput
# cd /home/oracle/dbinput
```

4. Create the input file in the directory. The following sample file will create a database configured to store backups in an existing bucket in Object Storage. For parameter descriptions, see Create Database Parameters.

```
{
  "object": "db",
  "action": "start",
  "operation": "createdb",
  "params": {
    "nodelist":              "",
    "dbname":                "exadb",
    "edition":               "EE_EP",
    "version":               "12.1.0.2",
    "adminPassword":         "<password>",
    "sid":                   "exadb",
    "pdbName":               "PDB1",
    "charset":               "AL32UTF8",
    "ncharset":              "AL16UTF16",
    "backupDestination":     "OSS",
    "cloudStorageContainer": "https://swiftobjectstorage.<region_
name>.oraclecloud.com/v1/mycompany/DBBackups",
    "cloudStorageUser":      "<name@example.com>",
    "cloudStoragePwd":       "<auth_token>"
  },
  "outputfile": "/home/oracle/createdb.out",
  "FLAGS": ""
}
```

5. Run the utility and specify the input file.

```
[root@dbsys ~]# /var/opt/oracle/dbaasapi/dbaasapi -i myinput.json
```

6. Check the output file and note the ID.

```
[root@dbsys ~]# cat /home/oracle/createdb.out
{
   "msg" : "",
   "object" : "db",
   "status" : "Starting",
   "errmsg" : "",
   "outputfile" : "/home/oracle/createdb.out",
   "action" : "start",
   "id" : "170",
   "operation" : "createdb",
   "logfile" : "/var/opt/oracle/log/gsa1/dbaasapi/db/createdb/1.log"
}
```

7. Create a JSON file to check the database creation status. Note the action of `"status"`. Replace the ID and the dbname with the values from the previous steps.

```
{
  "object": "db",
  "action": "status",
  "operation": "createdb",
  "id": 170,
  "params": {
    "dbname": "exadb"
  },
  "outputfile": "/home/oracle/createdb.out",
  "FLAGS": ""
}
```

8. Run the utility with the status file as input and then check the utility output.
   Rerun the status action regularly until the response indicates that the operation succeeded or failed.

```
[root@dbsys ~]# /var/opt/oracle/dbaasapi/dbaasapi -i db_status.json

[root@dbsys ~]# cat /home/oracle/createdb.out

{
```

```
   "msg" : "Sync sqlnet file...[done]\\n##Done executing tde\\nWARN: Could not register elogger_
parameters: elogger.pm::_init: /var/opt/oracle/dbaas_acfs/events does not exist\\n##Invoking
assistant bkup\\nUsing cmd : /var/opt/oracle/ocde/assistants/bkup/bkup -out
/var/opt/oracle/ocde/res/bkup.out -sid=\"exadb1\" -reco_grp=\"RECOC1\" -
hostname=\"ed1db01.data.customer1.oraclevcn.com\" -oracle_
home=\"/u02/app/oracle/product/12.1.0/dbhome_5\" -dbname=\"exadb\" -dbtype=\"exarac\" -
exabm=\"yes\" -edition=\"enterprise\" -bkup_cfg_files=\"no\" -acfs_vol_
dir=\"/var/opt/oracle/dbaas_acfs\" -bkup_oss_url=\"bkup_oss_url\" -bkup_oss_user=\"bkup_oss_
user\" -version=\"12102\" -oracle_base=\"/u02/app/oracle\" -firstrun=\"no\" -action=\"config\" -
bkup_oss=\"no\" -bkup_disk=\"no\" -data_grp=\"DATAC1\" -action=config \\n\\n##Done executing
bkup\\nWARN: Could not register elogger_parameters: elogger.pm::_init: /var/opt/oracle/dbaas_
acfs/events does not existRemoved all entries from creg file : /var/opt/oracle/creg/exadb.ini
matching passwd or decrypt_key\\n\\n#### Completed OCDE Successfully ####\\nWARN: Could not
register elogger_parameters: elogger.pm::_init: /var/opt/oracle/dbaas_acfs/events does not
exist",
   "object" : "db",
   "status" : "Success",
   "errmsg" : "",
   "outputfile" : "/home/oracle/createdb_exadb.out",
   "action" : "start",
   "id" : "170",
   "operation" : "createdb",
   "logfile" : "/var/opt/oracle/log/exadb/dbaasapi/db/createdb/170.log"
}
```


CREATE DATABASE PARAMETERS

Use the following parameters to create a database.

Parameter	Description
object	The value `"db"`.
action	The value `"start"`.
operation	The value `"createdb"`.
nodelist	The value `""` (an empty string). The database will be created across all nodes in the cluster.

Parameter	Description
dbname	The database name, in quotes.
edition	The value "EE_EP". (Only Enterprise Edition - Extreme Performance is supported .)
version	The database version as 18.0.0.0, 12.2.0.1, 12.1.0.2, or 11.2.0.4, in quotes.
adminPassword	The administrator (SYS and SYSTEM) password to use for the new database, in quotes. The password must be nine to thirty characters and contain at least two uppercase, two lowercase, two numeric, and two special characters. The special characters must be _, #, or -.
sid	The SID of the database, in quotes.
pdbName	The name of the pluggable database, in quotes.

Parameter	Description
charset	The database character set, in quotes. Allowed values AL32UTF8, AR8ADOS710, AR8ADOS720, AR8APTEC715, AR8ARABICMACS, AR8ASMO8X, AR8ISO8859P6, AR8MSWIN1256, AR8MUSSAD768, AR8NAFITHA711, AR8NAFITHA721, AR8SAKHR706, AR8SAKHR707, AZ8ISO8859P9E, BG8MSWIN, BG8PC437S, BLT8CP921, BLT8ISO8859P13, BLT8MSWIN1257, BLT8PC775, BN8BSCII, CDN8PC863, CEL8ISO8859P14, CL8ISO8859P5, CL8ISOIR111, CL8KOI8R, CL8KOI8U, CL8MACCYRILLICS, CL8MSWIN1251, EE8ISO8859P2, EE8MACCES, EE8MACCROATIANS, EE8MSWIN1250, EE8PC852, EL8DEC, EL8ISO8859P7, EL8MACGREEKS, EL8MSWIN1253, EL8PC437S, EL8PC851, EL8PC869, ET8MSWIN923, HU8ABMOD, HU8CWI2, IN8ISCII, IS8PC861, IW8ISO8859P8, IW8MACHEBREWS, IW8MSWIN1255, IW8PC1507, JA16EUC, JA16EUCTILDE, JA16SJIS, JA16SJISTILDE, JA16VMS, KO16KSCCS, KO16MSWIN949, LA8ISO6937, LA8PASSPORT, LT8MSWIN921, LT8PC772, LT8PC774, LV8PC1117, LV8PC8LR, LV8RST104090, N8PC865, NE8ISO8859P10, NEE8ISO8859P4, RU8BESTA, RU8PC855, RU8PC866, SE8ISO8859P3, TH8MACTHAIS, TH8TISASCII, TR8DEC, TR8MACTURKISHS, TR8MSWIN1254, TR8PC857, US7ASCII, US8PC437, UTF8, VN8MSWIN1258, VN8VN3, WE8DEC, WE8DG, WE8ISO8859P15, WE8ISO8859P9, WE8MACROMAN8S, WE8MSWIN1252, WE8NCR4970, WE8NEXTSTEP, WE8PC850, WE8PC858, WE8PC860, WE8ROMAN8, ZHS16CGB231280, ZHS16GBK, ZHT16BIG5, ZHT16CCDC, ZHT16DBT, ZHT16HKSCS, ZHT16MSWIN950, ZHT32EUC, ZHT32SOPS, ZHT32TRIS

Parameter	Description
ncharset	The database national character set. The value `AL16UTF16` or `UTF8`, in quotes.
backupDestination	The database backup destination, in quotes. You can configure the following backup destinations. `NONE` No backup destination is configured. `DISK` Configure database backups to the local disk Fast Recovery Area. `OSS` Configure database backups to an existing bucket in the Oracle Cloud Infrastructure Object Storage service. You must specify all the `cloudStorage` parameters. `BOTH` Configure database backups to both local disk and an existing bucket in Object Storage. You must specify all the `cloudStorage` parameters. For example: `"backupDestination":"BOTH"`

Parameter	Description
cloudStorageContainer= *<swift_url>*	Required if you specify a backup destination of `OSS` or `BOTH`. The Object Storage URL, your Oracle Cloud Infrastructure tenant, and an existing bucket in the object store to use as the backup destination, in the following format: `https://swiftobjectstorage.<region_name>.oraclecloud.com/v1/<tenant>/<bucket>` See Regions and Availability Domains to look up the region name string. For example: `"cloudStorageContainer":"https://swiftobjectstorage.<region_name>.oraclecloud.com/v1/<company_name>/DBBackups"`
cloudStorageUser= *<user_name>*	Required if you specify a backup destination of `OSS` or `BOTH`. The user name for the Oracle Cloud Infrastructure user account, for example: `"cloudStorageUser":"name@company.com"` This is the user name you use to sign in to the Console. The user name must be a member of the Administrators group, as described in Prerequisites.

Parameter	Description
cloudStoragePwd= *<auth_token>*	Required if you specify a backup destination of `OSS` or `BOTH`. The auth token generated by using the Console or IAM API, in quotes, for example: `"cloudStoragePwd":"<auth_token>"` For more information, see Managing User Credentials. This is **not** the password for the Oracle Cloud Infrastructure user.
outputfile	The absolute path for the output of the request, for example, `"outputfile":"/home/oracle/createdb.out"`.
FLAGS	The value `""` (an empty string).

Deleting a Database

Oracle recommends that you create a final backup before you delete any production (non-test) database. See Backing Up an Exadata Database to learn how to back up an Exadata database.

1. SSH to a compute node in the Exadata DB system.

```
ssh -i <private_key_path> opc@<node_ip_address>
```

2. Log in as opc and then sudo to the root user.

```
login as: opc

[opc@dbsys ~]$ sudo su -
```

3. Make a directory for the input file and change to the directory.

```
[root@dbsys ~]# mkdir -p /home/oracle/dbinput


# cd /home/oracle/dbinput
```

4. Create the input file in the directory and specify the database name to delete and an output file. For more information, see Delete Database Parameters .

```
{
  "object":  "db",
  "action":  "start",
  "operation": "deletedb",
  "params": {
    "dbname": "exadb"
  },
   "outputfile":   "/home/oracle/delete_exadb.out",
   "FLAGS": ""
}
```

5. Run the utility and specify the input file.

```
[root@dbsys ~]# /var/opt/oracle/dbaasapi/dbaasapi -i myinput.json
```

6. Check the output file and note the ID.

```
[root@ed1db01 ~]# cat /home/oracle/delete_exadb.out


{
   "msg" : "",
   "object" : "db",
   "status" : "Starting",
   "errmsg" : "",
   "outputfile" : "/home/oracle/deletedb.out",
   "action" : "start",
   "id" : "17",
   "operation" : "deletedb",
   "logfile" : "/var/opt/oracle/log/exadb/dbaasapi/db/deletedb/17.log"
}
```

7. Create a JSON file to check the database deletion status. Note the action of "status" in the sample file below. Replace the ID and the dbname with the values from the previous steps.

```
{
  "object": "db",
  "action": "status",
  "operation": "deletedb",
```

```
  "id": 17,
  "params": {
    "dbname": "exadb"
  },
  "outputfile": "/home/oracle/deletedb.out",
  "FLAGS": ""
}
```

8. Run the utility with the status file as input and then check the utility output.
   Rerun the status action regularly until the response indicates that the operation succeeded.

```
[root@dbsys ~]#  /var/opt/oracle/dbaasapi/dbaasapi -i db_status.json

[root@dbsys ~]# cat /home/oracle/deletedb.out

{
   "msg" : "Using cmd : su - root -c \"/var/opt/oracle/ocde/assistants/dg/dgcc -dbname exadb -
action delete\" \\n\\n##Done executing dg\\nWARN: Could not register elogger_parameters:
elogger.pm::_init: /var/opt/oracle/dbaas_acfs/events does not exist\\n##Invoking assistant
bkup\\nUsing cmd : /var/opt/oracle/ocde/assistants/bkup/bkup -out
/var/opt/oracle/ocde/res/bkup.out -bkup_oss_url=\"bkup_oss_url\" -bkup_daily_time=\"0:13\" -bkup_
oss_user=\"bkup_oss_user\" -dbname=\"exadb\" -dbtype=\"exarac\" -exabm=\"yes\" -firstrun=\"no\" -
action=\"delete\" -bkup_cfg_files=\"no\" -bkup_oss=\"no\" -bkup_disk=\"no\" -action=delete
\\n\\n##Done executing bkup\\nWARN: Could not register elogger_parameters: elogger.pm::_init:
/var/opt/oracle/dbaas_acfs/events does not exist\\n##Invoking assistant dbda\\nUsing cmd :
/var/opt/oracle/ocde/assistants/dbda/dbda -out /var/opt/oracle/ocde/res/dbda.out -em=\"no\" -pga_
target=\"2000\" -dbtype=\"exarac\" -sga_target=\"2800\" -action=\"delete\" -build=\"no\" -
nid=\"no\" -dbname=\"exadb\" -action=delete \\n",
   "object" : "db",
   "status" : "InProgress",
   "errmsg" : "",
   "outputfile" : "/home/oracle/deletedb.out",
   "action" : "start",
   "id" : "17",
   "operation" : "deletedb",
   "logfile" : "/var/opt/oracle/log/exadb/dbaasapi/db/deletedb/17.log"
}
```

DELETE DATABASE PARAMETERS

Use the following parameters to delete a database.

Parameter	Description
object	The value `"db"`.
action	The value `"start"`.
operation	The value `"deletedb"`.
dbname	The database name, in quotes.
outputfile	The absolute path for the output of the request, for example, `"/home/oracle/deletedb.out"`.
FLAGS	The value `""` (an empty string).

Backing Up an Exadata Database

You can back up databases on an Exadata DB system to an existing bucket in the Oracle Cloud Infrastructure Object Storage service and to the local disk Fast Recovery Area.

This topic explains how to:

- Create a backup configuration file that indicates the backup destination, when the backup should run, and how long backups are retained. If the backup destination is Object Storage, the file also contains the credentials to access the service.
- Associate the backup configuration file with a database. The database will be backed up as scheduled, or you can create an on-demand backup.



Note

You must update the cloud-specific tooling on all the compute nodes in your Exadata DB system before performing the following procedures. For more information, see Updating an Exadata DB System.

Prerequisites

- The Exadata DB system requires access to the Oracle Cloud Infrastructure Object Storage service. Oracle recommends using a service gateway with the VCN to enable this access. For more information, see Network Setup for Exadata DB Systems. In that topic, pay particular attention to:
  - Service Gateway for the VCN
  - Node Access to Object Storage: Static Route
  - Rule for Backup Subnet's Custom Security List
- An existing Object Storage bucket to use as the backup destination. You can use the Console or the Object Storage API to create the bucket. For more information, see Managing Buckets.
- An auth token generated by Oracle Cloud Infrastructure. You can use the Console or the IAM API to generate the password. For more information, see Working with Auth Tokens.
- The user name specified in the backup configuration file must have tenancy-level access to Object Storage. An easy way to do this is to add the user name to the Administrators group. However, that allows access to *all* of the cloud services. Instead, an administrator should create a policy like the following that limits access to only the required resources in Object Storage for backing up and restoring the database:

  ```
  Allow group <group_name> to manage objects in compartment <compartment_name> where
  target.bucket.name = '<bucket_name>'

  Allow group <group_name> to read buckets in compartment <compartment_name>
  ```

  For more information about adding a user to a group, see Managing Groups. For more information about policies, see Getting Started with Policies.

Default Backup Configuration

The backup configuration follows a set of Oracle best-practice guidelines:

- Full (level 0) backup of the database followed by rolling incremental (level 1) backups on a seven-day cycle (a 30-day cycle for the Object Storage destination).
- Full backup of selected system files.
- Automatic backups daily at a specific time set during the database deployment creation process.

Retention period:

- Both Object Storage and local storage: 30 days, with the 7 most recent days' backups available on local storage.
- Object Storage only: 30 days.
- Local storage only: Seven days.

Encryption:

- Both Object Storage and local storage: All backups to cloud storage are encrypted.
- Object Storage only: All backups to cloud storage are encrypted.

Managing Backups

To create a backup configuration file



Important

The following procedure must be performed on the first compute node in the Exadata DB system. To determine the first compute node, connect to any compute node as the `grid` user and execute the following command:

```
$ $ORACLE_HOME/bin/olsnodes -n
```

The first node has the number 1 listed beside the node name.

1. SSH to the first compute node in the Exadata DB system.

```
ssh -i <private_key_path> opc@<node_1_ip_address>
```

2. Log in as opc and then sudo to the root user.

```
login as: opc

[opc@dbsys ~]$ sudo su -
```

3. Create a new backup configuration file in `/var/opt/oracle/ocde/assistants/bkup/` as shown in the sample configuration file below. This example uses the file name `bkup.cfg`, but you can provide your own file name. The following file schedules a backup to both local storage and an existing bucket in Object Storage.
   The parameters are described below this procedure.

```
[root@dbsys ~]# cd /var/opt/oracle/ocde/assistants/bkup/

vi bkup.cfg
bkup_disk=yes
bkup_oss=yes
bkup_oss_url=https://swiftobjectstorage.<region_name>.oraclecloud.com/v1/companyabc/DBBackups
bkup_oss_user=name@example.com
bkup_oss_passwd=<auth_token>
bkup_oss_recovery_window=7
bkup_daily_time=06:45
```

4. Change the permissions of the file.

```
[root@dbsys bkup]# chmod 600 bkup.cfg
```

5. Use the following command to install the backup configuration, configure the credentials, schedule the backup, and associate the configuration with a database name.

```
[root@dbsys bkup]# ./bkup -cfg bkup.cfg -dbname=<database_name>
```

   The backup is scheduled via cron and can be viewed at `/etc/crontab`.

When the scheduled backup runs, you can check its progress with the following command.

```
[root@dbsys bkup]# /var/opt/oracle/bkup_api/bkup_api bkup_status
```

The backup configuration file parameters are described in the following table:

Parameter	Description
bkup_disk=[yes|no]	Whether to back up locally to disk (Fast Recovery Area).
bkup_oss=[yes|no]	Whether to back up to Object Storage. If yes, you must also provide the parameters bkup_oss_url, bkup_oss_user, bkup_oss_passwd, and bkup_oss_recovery_window.

Parameter	Description
bkup_oss_url=*<swift_url>*	Required if bkup_oss=yes. The Object Storage URL including the tenant and bucket you want to use. The URL is: `https://swiftobjectstorage.<region_name>.oraclecloud.com/v1/<tenant>/<bucket>` where *<tenant>* is the *lowercase* tenant name (even if it contains uppercase characters) that you specify when signing in to the Console and *<bucket>* is the name of the existing bucket you want to use for backups. See Regions and Availability Domains to look up the region name string.

Parameter	Description
bkup_oss_user=*<oci_user_name>*	Required if bkup_oss=yes. The user name for the Oracle Cloud Infrastructure user account, which takes the form `name@example.com`. The user must be a member of the Administrators group, as described in Prerequisites. This is the user name you use to sign in to the Console.
bkup_oss_passwd=*<auth_token>*	Required if bkup_oss=yes. The auth token generated by using the Console or IAM API, as described in Prerequisites. This is **not** the password for the Oracle Cloud Infrastructure user.

Parameter	Description
bkup_oss_recovery_window=*n*	Required if bkup_oss=yes. The number of days for which backups and archived redo logs are maintained in the Object Storage bucket. Specify 1 to 30 days.
bkup_daily_time=*hh:mm*	The time at which the daily backup is scheduled, specified in hours and minutes (hh:mm), in 24-hour format.

To create an on-demand backup

You can use the `bkup_api` utility to create an on-demand backup of a database.



Warning

Oracle recommends that you avoid supplying values that include confidential information when you use the `bkup_api` command.

1. SSH to the first compute node in the Exadata DB system.

   ```
   ssh -i <private_key_path> opc@<node_1_IP_address>
   ```

   To determine the first compute node, connect to any compute node as the `grid` user and execute the following command:

   ```
   $ $ORACLE_HOME/bin/olsnodes -n
   ```

The first node has the number 1 listed beside the node name.

2. Log in as opc and then sudo to the root user.

```
login as: opc

[opc@dbsys ~]$ sudo su -
```

3. You can let the backup follow the current retention policy, or you can create a long-term backup that persists until you delete it:
   - To create a backup that follows the current retention policy, enter the following command:

     ```
     # /var/opt/oracle/bkup_api/bkup_api bkup_start --dbname=<database_name>
     ```

   - To create a long-term backup, enter the following command:

     ```
     # /var/opt/oracle/bkup_api/bkup_api bkup_start --keep --dbname=<database_name>
     ```

4. Exit the root-user command shell and disconnect from the compute node:

```
# exit
$ exit
```

By default, the backup is given a timestamp-based tag. To specify a custom backup tag, add the `--tag` option to the `bkup_api` command; for example, to create a long-term backup with the tag "monthly", enter the following command:

```
# /var/opt/oracle/bkup_api/bkup_api bkup_start --keep --tag=monthly
```

After you enter a `bkup_api bkup_start` command, the `bkup_api` utility starts the backup process, which runs in the background. To check the progress of the backup process, enter the following command:

```
# /var/opt/oracle/bkup_api/bkup_api bkup_status --dbname=<database_name>
```

To remove the backup configuration

A backup configuration can contain the credentials to access the Object Storage bucket. For this reason, you might want to remove the file after successfully configuring the backup.

```
[root@dbsys bkup]# rm bkup.cfg
```

To delete a local backup

To delete a backup of a database deployment on the Exadata DB system, use the `bkup_api` utility.

1. Connect to the first compute node in your Exadata DB system as the `opc` user.
   To determine the first compute node, connect to any compute node as the `grid` user and execute the following command:

   ```
   $ $ORACLE_HOME/bin/olsnodes -n
   ```

   The first node has the number 1 listed beside the node name.
2. Start a root-user command shell:

   ```
   $ sudo -s#
   ```
3. List the available backups:

   ```
   # >/var/opt/oracle/bkup_api/bkup_api recover_list --dbname=<database_name>
   ```

   where `dbname` is the database name for the database that you want to act on.
   A list of available backups is displayed.
4. Delete the backup you want:

   ```
   # /var/opt/oracle/bkup_api/bkup_api bkup_delete --bkup=<backup_tag> --dbname=<database_name>
   ```

   where `backup_tag` is the tag of the backup you want to delete.
5. Exit the root-user command shell:

   ```
   # exit$
   ```

To delete a backup in Object Storage

Use the RMAN `delete backup` command to delete a backup from the Object Store.

WHAT NEXT?

If you used Object Storage as a backup destination, you can display the backup files in your bucket in the Console on the **Storage** page, by selecting **Object Storage**.

You can manually restore a database backup by using the RMAN utility. For information about using RMAN, see the *Oracle Database Backup and Recovery User's Guide* for Release 18.1, 12.2, 12.1, or 11.2.

Recovering an Exadata Database

Exadata DB systems provide a backup feature that backs up the Oracle database associated with a database deployment. This feature is built over Oracle Recovery Manager (RMAN).

You can manually restore a database backup by using the RMAN utility. For information about using RMAN, see the *Oracle Database Backup and Recovery User's Guide* for Release 18.1, 12.2, 12.1, or 11.2.

DB System Time Zone



Note

This topic applies only to Exadata DB systems.

The Time Zone field in the Console and in the API allows you to launch an Exadata DB system with a time zone other than UTC (the default). Although UTC is the recommended time zone to use, having a common time zone for your database clients and application hosts can simplify management and troubleshooting for the database administrator.

The time zone that you specify when you create the DB system applies to the host and to the Oracle Grid Infrastructure, and controls the time zone of the database log files. The time zone of the database itself is not affected, however, the database's time zone affects only the timestamp datatype. By default, it is set to UTC and although you can change it manually, Oracle recommends that you keep it as UTC to avoid data conversion and improve

performance when data is transferred among databases. This is especially important for distributed databases, replication, and exporting and importing.

Time Zone Options

Whether you use the Console or the API, the time zone options you can select from are represented in the named region format, for example, *America/Los_Angeles*. The Console allows you to select UTC, the time zone detected in your browser (if your browser supports time zone detection), or an alternate time zone.

To specify an alternate time zone (the **Select Another Time Zone** option), you first select a value in the Time Zone Prefix field to narrow the list of time zones from which to select in the Time Zone Suffix field. In the *America/Los_Angeles* example, *America* is the time zone prefix and *Los_Angeles* is the time zone suffix. The items you see in the Time Zone Prefix and Time Zone Suffix fields roughly correlate with the time zones supported in both the Java.util.TimeZone class and on the Linux operating system. If you do not see the time zone you are looking for, try selecting "Miscellaneous" as the time zone prefix.



Tip

If you are using the API and would like to see a list of supported time zones, you can examine the time zone options in the Console. These options appear in the **Launch DB System** dialog when you show advanced options after you select an Exadata DB system shape.

Changing Time Zones After Provisioning

Follow these steps if you need to change the time zone of the Exadata DB system, Oracle Grid Infrastructure, or database, after you launch the DB system:

To change the time zone of the DB system host

1. Log on to the host system as `root`.
2. Stop the CRS stack on all of the Exadata compute nodes.

```
#Grid_Home/bin/crsctl stop crs
```

3. Run the following commands to check the current time zone and to change it to the time zone you choose:

```
$ cat /etc/sysconfig/clock
ZONE="America/New_York"
$ cp -p /etc/sysconfig/clock /etc/sysconfig/clock.20160629

$ vi /etc/sysconfig/clock
ZONE="Europe/Berlin"

$ date
Wed Jun 29 10:35:17 EDT 2016
$ ln -sf /usr/share/zoneinfo/Europe/Berlin /etc/localtime
$ date
Wed Jun 29 16:35:27 CEST 2016
```

   In this example, the time zone was changed from *America/New_York* to *Europe/Berlin*.



Tip

To see a list of valid time zones on the host, you can run the `ls -l /usr/share/zoneinfo` command.

4. (Optional) Verify that `/opt/oracle.cellos/cell.conf` indicates the correct time zone. Using our example, the time zone entry in this file would be `<Timezone>Europe/Berlin</Timezone>`.
5. Restart the CRS stack on all of the Exadata compute nodes.

```
#Grid_Home/bin/crsctl start crs
```

To change the time zone of the Oracle Grid Infrastructure

The time zone of the Oracle Grid Infrastructure determines the time zone of the database log files. You can change this time zone by updating the `TZ` property in the `GRID_HOME/crs/install/s_crsconfig_<node_name>_env.txt` configuration file.

1. Ensure that you are logged onto the host as `root` and that the CRS stack is stopped on all of the Exadata compute nodes. See To change the time zone of the DB system host.
2. Inspect the current time zone value in the `GRID_HOME/crs/install/s_crsconfig_<node_name>_env.txt` file.

```
$ cat /u01/app/12.1.0.2/grid/crs/install/s_crsconfig_node1_env.txt
################################################################################
#This file can be used to set values for the NLS_LANG and TZ environment
#variables and to set resource limits for Oracle Clusterware and
#Database processes.
#1. The NLS_LANG environment variable determines the language and
#   characterset used for messages. For example, a new value can be
#   configured by setting NLS_LANG=JAPANESE_JAPAN.UTF8
#2. The Time zone setting can be changed by setting the TZ entry to
#   the appropriate time zone name. For example, TZ=America/New_York
#3. Resource limits for stack size, open files and number of processes
#   can be specified by modifying the appropriate entries.
#
#Do not modify this file except as documented above or under the
#direction of Oracle Support Services.
################################################################################
TZ=UTC
NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P1
CRS_LIMIT_STACK=2048
CRS_LIMIT_OPENFILE=65536
CRS_LIMIT_NPROC=16384
TNS_ADMIN=
```

   In this example, the time zone is set to UTC.
3. Modify the time zone value, as applicable. Perform this task for all nodes in the cluster.

4. Restart the CRS stack on all of the Exadata compute nodes.

```
#Grid_Home/bin/crsctl start crs
```

For more information about changing the time zone of the Grid Infrastructure, see How To Change Timezone for Grid Infrastructure (Doc ID 1209444.1).

To change the time zone of a database

Use the `ALTER DATABASE SET TIME_ZONE` command to change the time zone of a database. This command takes either a named region such as America/Los_Angeles or an absolute offset from UTC.

This example sets the time zone to UTC:

```
ALTER DATABASE SET TIME_ZONE = '+00:00';
```

You must restart the database for the change to take effect. For more information, see Setting the Database Time Zone.

Bare Metal and Virtual Machine DB Systems

Oracle Cloud Infrastructure offers 1-node DB systems on either bare metal or virtual machines, and 2-node RAC DB systems on virtual machines.

You can manage these systems by using the Console, the API, the Oracle Cloud Infrastructure CLI, the Database CLI (DBCLI), Enterprise Manager, Enterprise Manager Express, or SQL Developer.



Note

This documentation is intended for Oracle database administrators and assumes familiarity with Oracle databases and tools. If you need additional information, see the product documentation available at http://docs.oracle.com/en/database/.

Supported Database Editions and Versions

All 1- node RAC DB systems support the following Oracle Database editions:

- Standard Edition
- Enterprise Edition
- Enterprise Edition - High Performance
- Enterprise Edition - Extreme Performance

2-node RAC DB systems require Oracle Enterprise Edition - Extreme Performance.

The supported database versions are:

- Oracle Database 18c
- Oracle Database 12c Release 2 (12.2)
- Oracle Database 12c Release 1 (12.1)
- Oracle Database 11g Release 2 (11.2)



Tip

Your DB system's operating system will periodically need to be updated, just as your Oracle Database software will need to be updated. Before attempting an OS update, be sure to read the information in Updating a DB System and back up your DB system's databases.

Availability of Older Database Versions for Virtual Machine DB Systems

For virtual machine DB systems, Oracle Cloud Infrastructure also supports the creation of DB systems using older database versions. For each shape, the latest version and recent prior versions of the release are available at provisioning.



Warning

If you need to launch your dbsystem with an older database version, see Critical Patch Updates for information on known security issues with your chosen database version. You will also need to analyze and patch known security issues for the operating system included with the older database version. See Securing Database for information on security best practices for databases in Oracle Cloud Infrastructure.

Bare Metal DB Systems

Bare metal DB systems consist of a single bare metal server running Oracle Linux 6.8, with locally attached NVMe storage. If the node fails, you can simply launch another system and restore the databases from current backups.

When you launch a bare metal DB system, you select a single Oracle Database Edition that applies to all the databases on that DB system. The selected edition cannot be changed. Each DB system can have multiple database homes, which can be different versions. Each database home can have only one database, which is the same version as the database home.

Shapes for Bare Metal DB Systems

When you launch a DB system, you choose a *shape*, which determines the resources allocated to the DB system. The available shapes for a bare metal DB system are:

- **BM.DenseIO2.52:** Provides a 1-node DB system (one bare metal server), with up to 52 CPU cores, 768 GB memory, and eight 6.4 TB locally attached NVMe drives (51.2 TB total) to the DB system.
- **BM.DenseIO1.36:** *Limited availability.* Provides a 1-node DB system (one bare metal server), with up to 36 CPU cores, 512 GB memory, and nine 3.2 TB locally attached NVMe drives (28.8 TB total) to the DB system.
  Note: BM.DenseO1.36 is available only to monthly universal credit customers existing on or before November 9th, 2018. This shape is available only in the us-phoenix-1, us-ashburn-1, and eu-frankfurt-1 regions.

Storage Considerations

The shape you choose for a bare metal DB system determines its total raw storage, but other options, like 2- or 3-way mirroring and the space allocated for data files, affect the amount of usable storage on the system. The following table shows how various configurations affect the usable storage for bare metal DB systems.

Shape	Raw Storage	Usable Storage with Normal Redundancy (2-way Mirroring)	Usable Storage with High Redundancy (3-way Mirroring)
BM.DenseIO2.52	51.2 TB NVMe	DATA 16 TB RECO 4 TB	DATA 9 TB RECO 2.3 TB
BM.DenseIO1.36 see note	28.8 TB NVMe	DATA 9.4 TB RECO 1.7 TB	DATA 5.4 TB RECO 1 TB

Note: BM.DenseIO1.36 availability is limited to monthly universal credit customers existing on or before November 9th, 2018, in the us-phoenix-1, us-ashburn-1, and eu-frankfurt-1 regions.

Virtual Machine DB Systems

There are two types of DB systems on virtual machines:

- A 1-node virtual machine DB system consists of one virtual machine.
- A 2-node virtual machine DB system consists of two virtual machines.

When you launch a virtual machine DB system, you select the Oracle Database Edition that applies to the database on that DB system. The selected edition cannot be changed. Unlike a bare metal DB system, a virtual machine DB system can have only a single database home, which in turn can have only a single database. The database can be a container database (CDB) with multiple pluggable databases (PDBs), if the edition is High Performance or Extreme Performance. The database will be the same version as the database home.

Virtual machine DB systems also differ from bare metal DB systems in the following ways:

- A virtual machine DB system database uses Oracle Cloud Infrastructure block storage instead of local storage. You specify a storage size when you launch the DB system, and you can scale up the storage as needed at any time.
- The number of CPU cores on an existing virtual machine DB system cannot be changed.

Fault Domain Considerations for 2-node Virtual Machine DB Systems

When you provision a 2-node RAC DB systems, the system assigns each node to a different fault domain by default. Using the **Advanced Options** link in the provisioning dialog, you can select the fault domain(s) to be used for your 2-node RAC DB systems and the system will assign the nodes to your selected fault domains. Oracle recommends that you place each node of a 2-node RAC DB system in a different fault domain. For more information on fault domains, see Fault Domains.

Shapes for Virtual Machine DB Systems

When you launch a DB system, you choose a *shape*, which determines the resources allocated to the DB system.

The following table shows the available shapes for a virtual machine DB system on X7.

Shape	CPU Cores	Memory
VM.Standard2.1	1	15 GB
VM.Standard2.2	2	30 GB
VM.Standard2.4	4	60 GB
VM.Standard2.8	8	120 GB
VM.Standard2.16	16	240 GB
VM.Standard2.24	24	320 GB

The following table shows the available shapes for a virtual machine DB system on X5.see note

Shape	CPU Cores	Memory
VM.Standard1.1see note	1	7 GB
VM.Standard1.2see note	2	14 GB

Shape	CPU Cores	Memory
VM.Standard1.4[see note]	4	28 GB
VM.Standard1.8[see note]	8	56 GB
VM.Standard1.16[see note]	16	112 GB

Note: X5-based shapes availability is limited to monthly universal credit customers existing on or before November 9th, 2018, in the us-phoenix-1, us-ashburn-1, and eu-frankfurt-1 regions.

Storage Options for Virtual Machine DB Systems

Virtual machine DB systems use Oracle Cloud Infrastructure block storage. The following table shows details of the storage options for a virtual machine DB system. Total storage includes available storage plus recovery logs.

Available Storage (GB)	Total Storage (GB)
256	712
512	968
1024	1480
2048	2656
4096	5116
6144	7572
8192	10032
10240	12488
12288	14944

Available Storage (GB)	Total Storage (GB)
14336	17404
16384	19860
18432	22320
20480	24776
22528	27232
24576	29692
26624	32148
28672	34608
30720	37064
32768	39520
34816	41980
36864	44436
38912	46896
40960	49352

For 2-node RAC virtual machine DB systems, storage capacity is shared between the nodes.

Network Setup for DB Systems



Note

This topic is not applicable to Exadata DB systems. For information on the network setup for an Exadata DB system, see Network Setup for Exadata DB Systems.

Before you set up a bare metal or virtual machine DB system, you must set up a virtual cloud network (VCN) and other Networking service components. This topic describes the recommended configuration for the VCN.

VCN and Subnets

To launch a DB system, you must have:

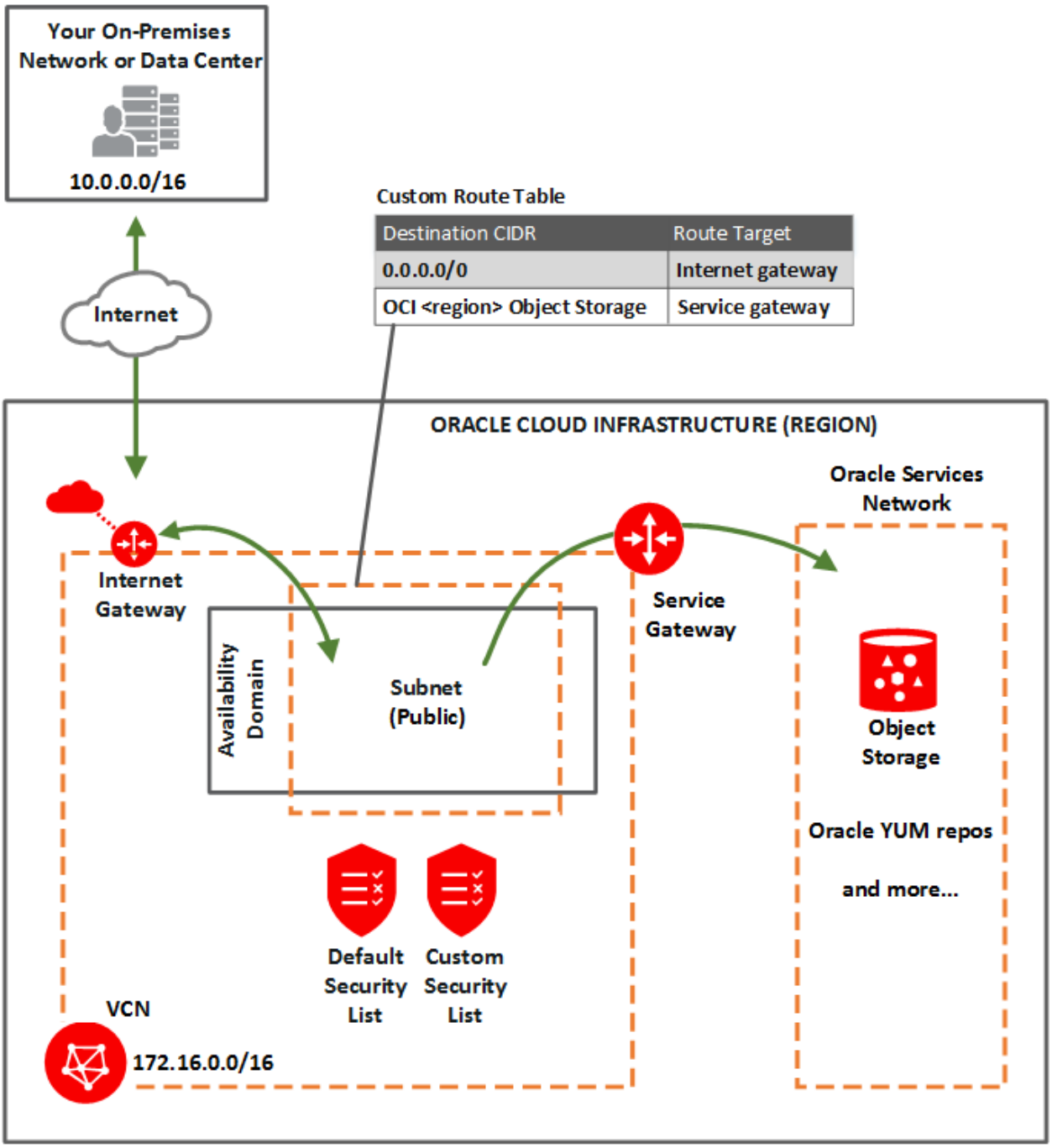
- A VCN in the region where you want the DB system
- At least one subnet in the VCN (either a public subnet or a private subnet)

In general, Oracle recommends using regional subnets, which span all availability domains in the region. For a bare metal or virtual machine DB system, either a regional subnet or AD-specific subnet works. For more information, see About Regional Subnets.

You will create a custom route table and security list for the subnet. More information follows about that.

Option 1: Public Subnet with Internet Gateway

This option can be useful when doing a proof-of-concept or development work. You can use this setup in production if you want to use an internet gateway with the VCN, or if you have services that run only on a public network and need access to the database. See the following diagram and description.


Your On-Premises Network or Data Center
10.0.0.0/16
Internet
Custom Route Table
Destination CIDR
Route Target
0.0.0.0/0
Internet gateway
OCI <region> Object Storage
Service gateway
ORACLE CLOUD INFRASTRUCTURE (REGION)
Oracle Services Network
Internet Gateway
Service Gateway
Availability Domain
Subnet (Public)
Object Storage
Oracle YUM repos
and more...
Default Security List
Custom Security List
VCN
172.16.0.0/16

You set up:

- Public subnet.
- Internet gateway.
- Service gateway to reach Object Storage for database backups and patching. Also see Option 1: Service Gateway Access Only to Object Storage.
- Route table: A custom route table for the subnet, with two rules:
  - A rule for 0.0.0.0/0, and target = internet gateway.
  - A rule for the service CIDR label called **OCI *<region>* Object Storage**, and target = the service gateway. Also see Option 1: Service Gateway Access Only to Object Storage.
- Security lists:
  - Default security list (with its default rules) for the subnet. **This security list contains several basic rules that are necessary for the DB system (such as ingress SSH access to the DB system, and general egress from the DB system).**
  - Custom security list for the subnet. For the rules to use, see Security Lists for the Subnet.

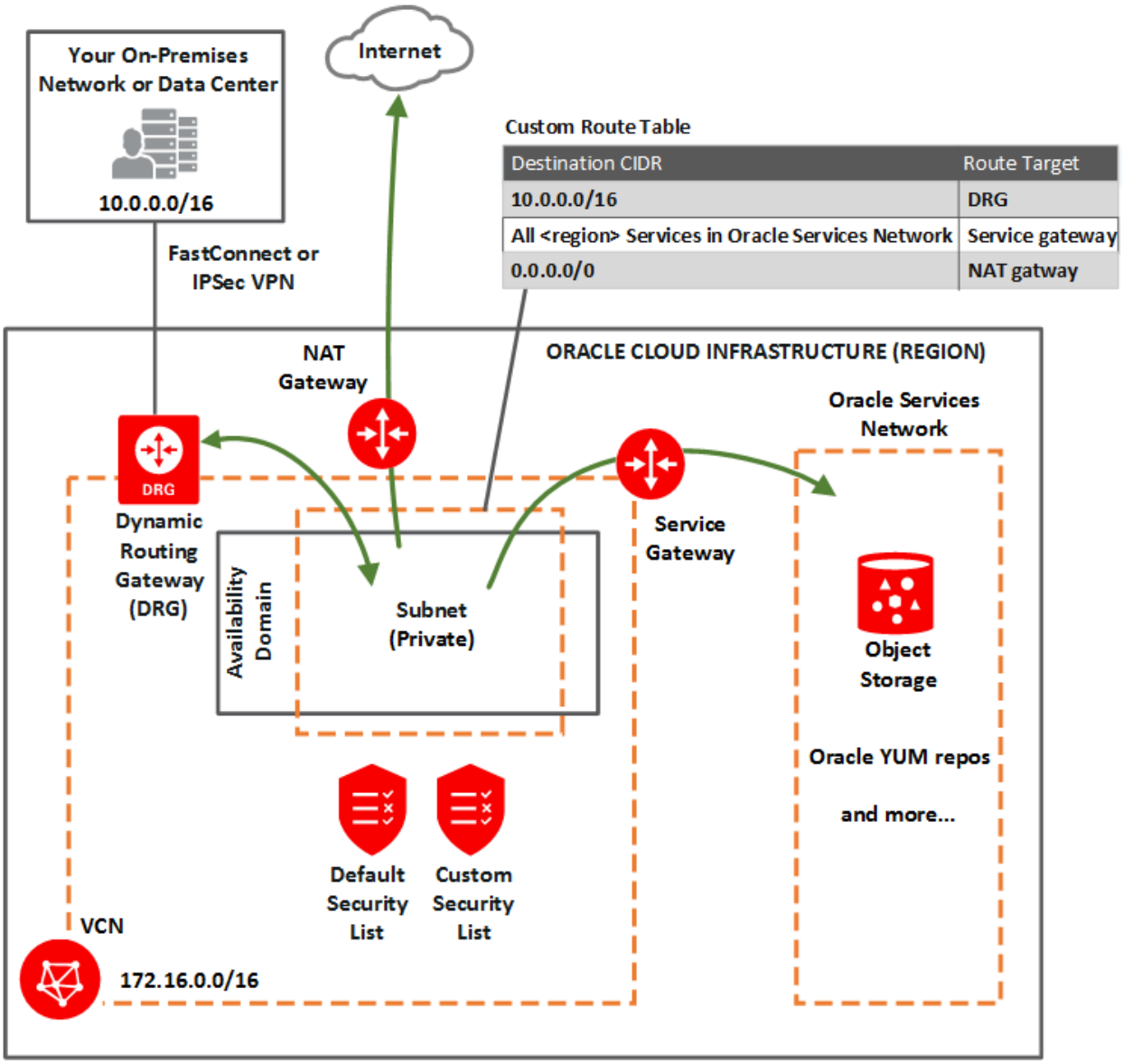


Important

See this known issue for information about configuring route rules with *service gateway* as the target on route tables associated with public subnets.

Option 2: Private Subnet

This option is recommended for a production system. The subnet is private and cannot be reached from the internet. See the following diagram and description.


Your On-Premises Network or Data Center
10.0.0.0/16
Internet
FastConnect or IPSec VPN
Custom Route Table
Destination CIDR
Route Target
10.0.0.0/16
DRG
All <region> Services in Oracle Services Network
Service gateway
0.0.0.0/0
NAT gatway
ORACLE CLOUD INFRASTRUCTURE (REGION)
NAT Gateway
Oracle Services Network
DRG
Dynamic Routing Gateway (DRG)
Service Gateway
Availability Domain
Subnet (Private)
Object Storage
Oracle YUM repos
and more...
Default Security List
Custom Security List
VCN
172.16.0.0/16

You set up:

- Private subnet.
- Gateways for the VCN:
  - Dynamic routing gateway (DRG), with a FastConnect or IPSec VPN to your on-premises network.
  - Service gateway to reach Object Storage for database backups and patching, and to reach Oracle YUM repos for OS updates. Also see Option 2: Service Gateway Access to Both Object Storage and YUM Repos.
  - NAT gateway (to reach public endpoints not supported by the service gateway).
- Route table: A custom route table for the subnet, with these rules:
  - A route for the on-premises network's CIDR, and target = DRG.
  - A rule for the service CIDR label called **All *<region>* Services in Oracle Services Network**, and target = the service gateway. Also see Option 2: Service Gateway Access to Both Object Storage and YUM Repos.
  - If you want to access the Oracle YUM repos through the NAT gateway, add a route rule for the regional YUM repo's public IP address, and target = the NAT gateway. If you just use the next rule only, the traffic to the YUM repo would still be routed to the service gateway, because the service gateway route is more specific than 0.0.0.0/0.
  - A rule for 0.0.0.0/0, and target = NAT gateway.
- Security lists:
  - Default security list (with its default rules) for the subnet. **This security list contains several basic rules that are necessary for the DB system (such as ingress SSH access to the DB system, and general egress from the DB system).**
  - Custom security list for the subnet. For the rules to use, see Security Lists for the Subnet.

Requirements for IP Address Space

If you're setting up DB systems (and thus VCNs) in more than one region, make sure the IP address space of the VCNs does not overlap.

The subnet you create for a bare metal or virtual machine DB system cannot overlap with 192.168.16.16/28, which is used by the Oracle Clusterware private interconnect on the database instance.

The following table lists the *minimum* required subnet size.



Tip

The Networking service reserves three IP addresses in each subnet. Allocating a larger space for the subnet than the minimum required (for example, at least /25 instead of /28) can reduce the relative impact of those reserved addresses on the subnet's available space.

DB System Type	# Required IP Addresses	Minimum Subnet Size
1-node bare metal or virtual machine	1 + 3 reserved in subnet = 4	/30 (4 IP addresses)
2-node RAC virtual machine	(2 addresses * 2 nodes) + 3 for SCANs + 3 reserved in subnet = 10	/28 (16 IP addresses)

VCN Creation Wizard: Not for Production

The Networking section of the Console includes a handy wizard that creates a VCN along with related resources. It can be useful if you just want to try launching an instance. However, the wizard automatically chooses the address ranges and creates public subnets and an internet gateway. You may not want this for your production network, so Oracle recommends you create the VCN and other resources individually yourself instead of using the wizard.

DNS for a Bare Metal DB System

For a bare metal DB system, you must ensure that the system can resolve the Swift endpoints (for backing up databases, patching, and updating the DB system's cloud tooling) and Oracle

YUM repo endpoints. Resolution of these endpoints is automatically covered by the Internet and VCN Resolver, which is the VCN's default choice for DNS in the VCN. For more information, see DNS in Your Virtual Cloud Network and also DHCP Options.

DNS FOR A VIRTUAL MACHINE DB SYSTEM

This information is applicable to 1-node or 2-node virtual machine DB systems.

For the node or nodes to communicate, the VCN must use the Internet and VCN Resolver. It enables hostname assignment to the nodes, and DNS resolution of those hostnames by resources in the VCN. It also enables resolution of the database's Single Client Access Names (SCANs). Lastly, it also enables resolution of the Swift endpoints and Oracle YUM repo endpoints. The Internet and VCN Resolver is the VCN's default choice for DNS in the VCN. For more information, see DNS in Your Virtual Cloud Network and also DHCP Options.

When you create the VCN, subnet, and DB system, you must carefully set the following identifiers, which are related to DNS in the VCN:

- VCN domain label
- Subnet domain label
- Hostname prefix for the DB system

These values make up the node's fully qualified domain name (FQDN):

<hostname_prefix><RAC_node_#>.*<subnet_domain_label>*.*<vcn_domain_label>*.oraclevcn.com

For RAC systems only, the Database service automatically appends a node number after the hostname prefix.

For example:

- Node 1: `dbsys1.ad1.acmevcniad.oraclevcn.com`
- Node 2: `dbsys2.ad1.acmevcniad.oraclevcn.com`

Requirement for the hostname prefix:

- Maximum 16 characters
- Cannot be the string *localhost*

Requirements for the VCN and subnet domain labels:

- Recommended maximum: 15 characters
- No hyphens
- Recommended: include the region name in the VCN's name, and include the availability domain name in the subnet's name

The FQDN has a maximum total limit of 63 characters, so make sure you set the VCN and subnet domain labels short enough to meet that requirement. Here is a safe general rule:

`<16_chars_max>#.<15_chars_max>.<15_chars_max>.oraclevcn.com`

The preceding maximums are not enforced when you create the VCN and subnets. However, the DB system deployment fails if the hostname prefix is greater than 16 characters or if the FQDN has more than 63 characters.

DNS: Between On-Premises Network and VCN

To enable the use of hostnames when on-premises hosts and VCN resources communicate with each other, you have two options:

- Set up an instance in the VCN to be a custom DNS server. For an example of an implementation of this scenario with the Oracle Terraform provider, see Hybrid DNS Configuration.
- Manage hostname resolution yourself manually.

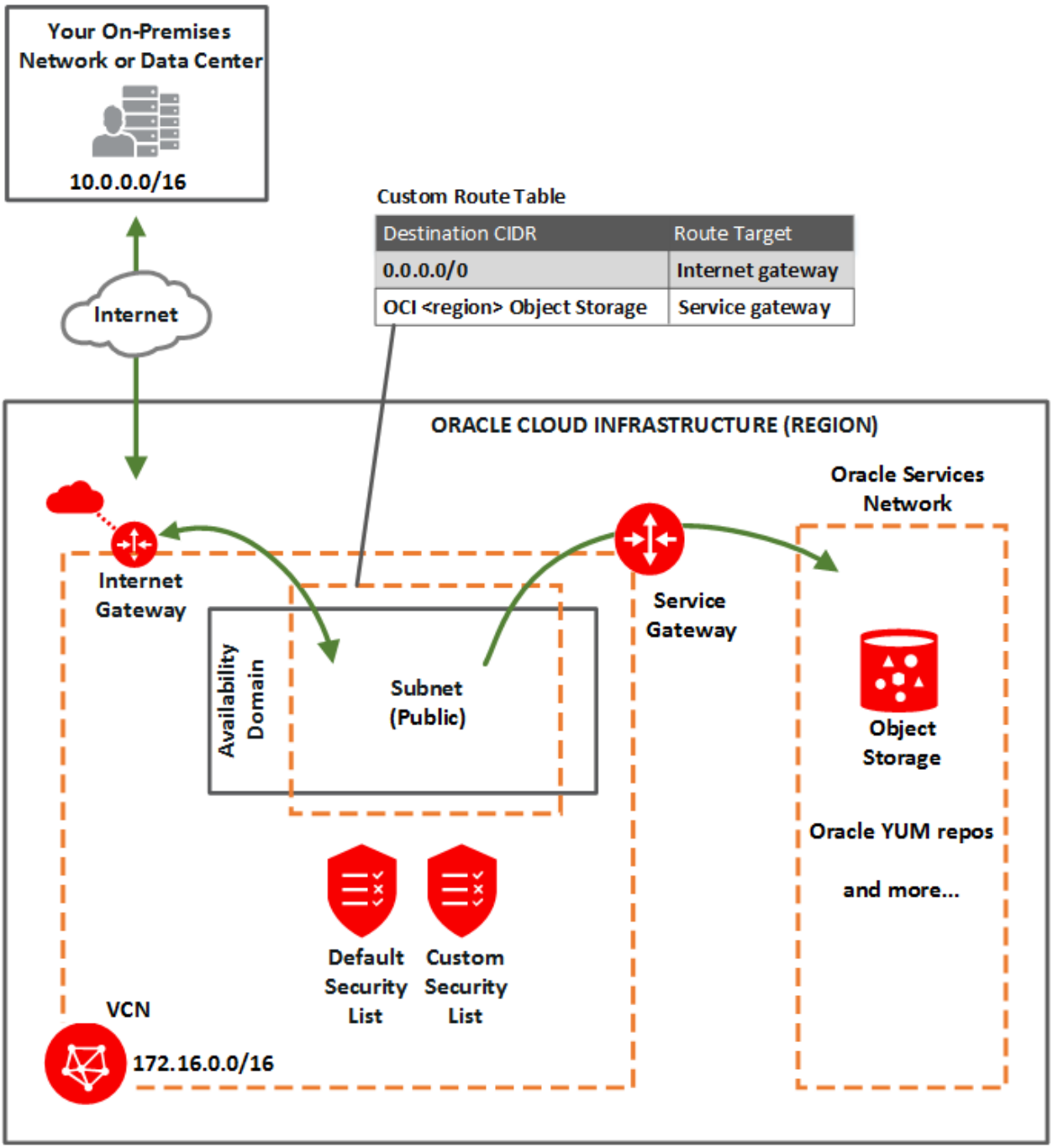
Service Gateway for the VCN

Your VCN needs access to both Object Storage (for backing up databases, patching, and updating the cloud tooling on a DB system) and Oracle YUM repos for OS updates.

Depending on whether you use option 1 or option 2 described previously, you use the service gateway in different ways. See the next two sections.

Option 1: Service Gateway Access Only to Object Storage

You configure the subnet to use the service gateway for access only to Object Storage. As a reminder, here's the diagram for option 1:


Your On-Premises Network or Data Center
10.0.0.0/16
Internet
Custom Route Table
Destination CIDR
Route Target
0.0.0.0/0
Internet gateway
OCI <region> Object Storage
Service gateway
ORACLE CLOUD INFRASTRUCTURE (REGION)
Oracle Services Network
Internet Gateway
Service Gateway
Availability Domain
Subnet (Public)
Object Storage
Oracle YUM repos
and more...
Default Security List
Custom Security List
VCN
172.16.0.0/16

In general, you must:

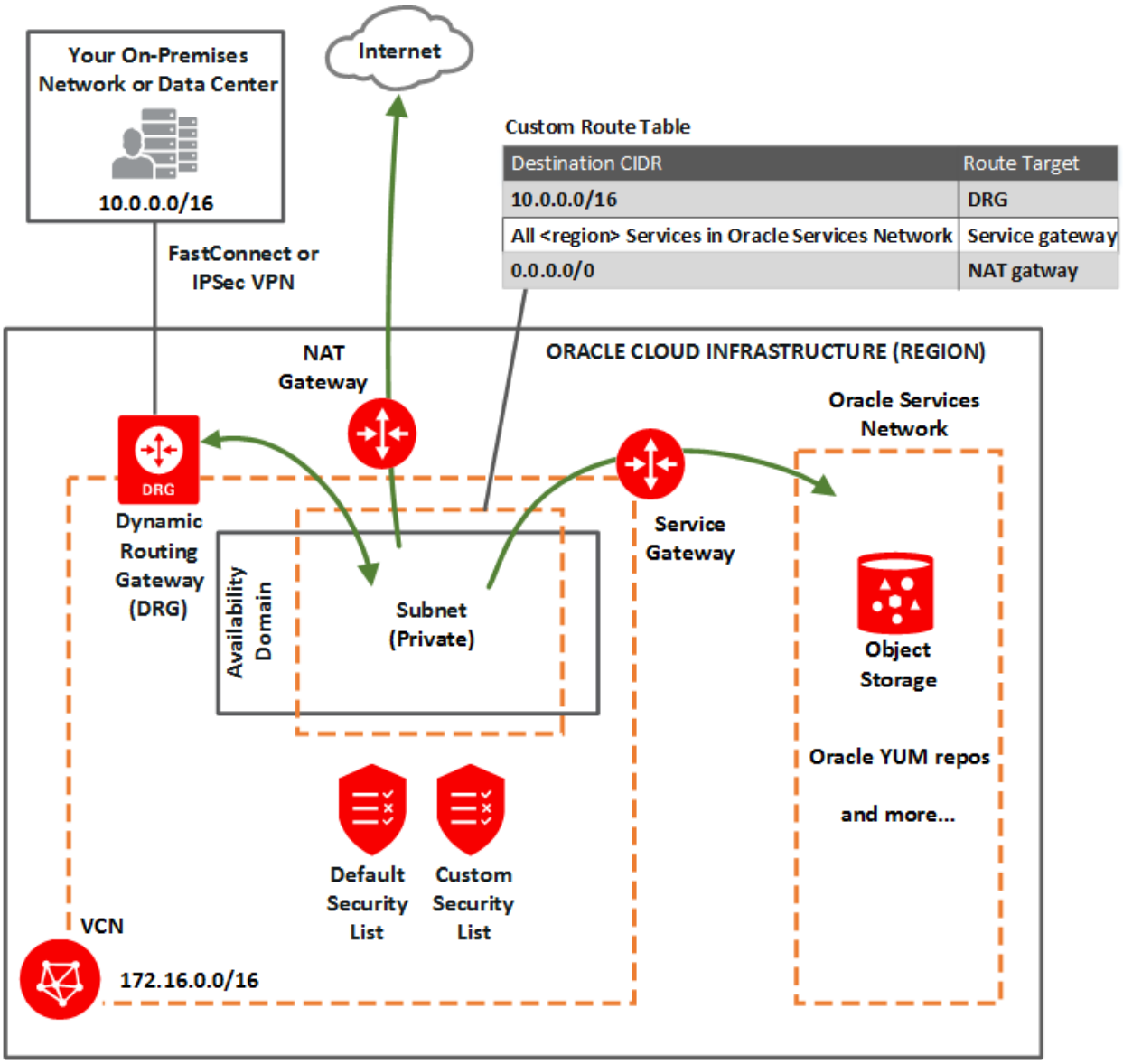
- Perform the tasks for setting up a service gateway on a VCN, and specifically enable the service CIDR label called **OCI *<region>* Object Storage**.
- In the task for updating routing, add a route rule to the subnet's custom route table. For the destination service, use **OCI *<region>* Object Storage** and target = the service gateway.
- In the optional task for updating security lists in the subnet, perform the task on the subnet's custom security list. Here you set up a security list rule with the destination service set to **OCI *<region>* Object Storage**. See Security Lists for the Subnet.

Option 2: Service Gateway Access to Both Object Storage and YUM Repos

You configure the subnet to use the service gateway for access to the Oracle Services Network, which includes both Object Storage and the Oracle YUM repos.

> **Important**
>
> See this known issue for information about accessing Oracle YUM services through the service gateway.

As a reminder, here's the diagram for option 2:



In general, you must:

- Perform the tasks for setting up a service gateway on a VCN, and specifically enable the service CIDR label called **All *<region>* Services in Oracle Services Network**.
- In the task for updating routing in the subnet, add a rule to the subnet's custom route table. For the destination service, use **All *<region>* Services in Oracle Services Network** and target = the service gateway.
- In the optional task for updating security lists in the subnet, perform the task on the subnet's custom security list. Here you set up a security list rule with the destination service set to **All *<region>* Services in Oracle Services Network**. See Security Lists for the Subnet.

Security Lists for the Subnet

Oracle recommends that you use at least two security lists with the subnet:

- VCN's default security list (with all its default rules). **This security list contains several basic rules that are necessary for the DB system (such as ingress SSH access to the DB system, and general egress from the DB system).**
- Custom security list with the rules in the next section.

You must create the custom security list yourself. The VCN's default security list is automatically created when you create the VCN. When you create a subnet, you can choose which security lists the subnets will use, or you can choose the security lists after the subnet is created.



Warning

Do not remove the default egress rule from the default security list. If you do, make sure to instead include a replacement egress rule in the subnet's custom security list, like so:

- **Stateless:** No (all rules must be stateful)
- **Destination Type:** CIDR
- **Destination CIDR:** 0.0.0.0/0
- **IP Protocol:** All

RULES FOR THE CUSTOM SECURITY LIST

Create the following security list rules in the subnet's custom security list and associate the security list with the subnet.

Ingress rule 1: Allows inbound SSH access

This rule enables SSH access to the DB system. It is redundant with a default ingress rule in the default security list. It is optional but recommended in case the default security list's rules are inadvertently changed.

- **Stateless:** No (all rules must be stateful)
- **Source Type:** CIDR
- **Source CIDR:** 0.0.0.0/0
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range:** 22

Ingress rule 2: Allows ONS and FAN traffic from within the VCN

This rule is recommended and enables the Oracle Notification Services (ONS) to communicate about Fast Application Notification (FAN) events.

- **Stateless:** No (all rules must be stateful)
- **Source Type:** CIDR
- **Source CIDR:** VCN's CIDR
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range:** 6200

Ingress rule 3: Allows SQL*NET traffic from within the VCN

This rule is for SQL*NET traffic and is required only if you need to enable client connections to the database.

- **Stateless:** No (all rules must be stateful)
- **Source Type:** CIDR
- **Source CIDR:** VCN's CIDR
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range:** 1521



Important

The preceding ingress rules 2 and 3 only cover connections initiated from within the VCN. If you have a client that resides *outside the VCN*, Oracle recommends setting up two *additional* similar rules that instead have



the **Source CIDR** set to the public IP address of the client.

Egress rule 1: Allows outbound SSH access

This rule enables SSH access between nodes in a 2-node DB system. It is redundant with the default egress rule in the default security list. It is optional but recommended in case the default security list's egress rule is inadvertently changed.

- **Stateless:** No (all rules must be stateful)
- **Destination Type:** CIDR
- **Destination CIDR:** 0.0.0.0/0
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range:** 22

Egress rule 2: Allows access to Object Storage and YUM repos

The next rule enables the DB system to communicate with Object Storage alone (for option 1), or with the Oracle Services Network, which includes both Object Storage and the Oracle YUM repos (for option 2). This rule is redundant with the default egress rule in the default security list. It is optional but recommended in case the default security list's rules are inadvertently changed.

- **Stateless:** No (all rules must be stateful)
- **Destination Type:** Service
- **Destination Service:**

    - For option 1, use the service CIDR label called **OCI *<region>* Object Storage**
    - For option 2, use the service CIDR label called **All *<region>* Services in Oracle Services Network**
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range:** 443 (HTTPS)

Managing Bare Metal and Virtual Machine DB Systems

This topic explains how to launch, start, stop, terminate, scale, manage licenses for, and check the status of a bare metal and virtual machine DB system, and set up DNS for a 1-node or 2-node RAC DB system.

When you launch a DB system using the Console, the API, or the CLI, the system is provisioned to support Oracle databases and an initial database is created based on the options you provide and some default options described later in this topic.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let database admins manage database systems lets the specified group do everything with databases and related Database resources.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for databases, see Details for the Database Service.

Prerequisites

You'll need the following items to launch any DB system.

- The public key, in OpenSSH format, from the key pair that you plan to use for connecting to the DB System via SSH. A sample public key, abbreviated for readability, is shown below.

  ```
  ssh-rsa AAAAB3NzaC1yc2EAAAABJQAA....lo/gKMLVM2xzc1xJr/Hc26biw3TXWGEakrK1OQ== rsa-key-20160304
  ```

  For more information, see Managing Key Pairs on Linux Instances.
- A correctly configured virtual cloud network (VCN) to launch the DB system in. Its related networking resources (gateways, route tables, security lists, DNS, and so on) must also be configured as necessary for the DB system. For more information, see Network Setup for DB Systems.
- If you plan to back up your DB system to Object Storage or to use the managed patching feature, Oracle recommends using a service gateway to enable access to Object Storage.
- For a 2-node RAC DB system, ensure that port 22 is open for both ingress and egress on the subnet, and that the security rules you create are stateful (the default), otherwise, the DB system might fail to provision successfully.

Default Options for the Initial Database

To simplify launching a DB system in the Console and when using the API, the following default options are used for the initial database and for any additional databases that you create. (Several advanced options such as Time Zone can be set when you can use the `dbcli` command line interface to create databases.)

- **Console Enabled:** False
- **Create Container Database:** False for version 11.2.0.4 databases. Otherwise, true.
- **Create Instance Only (for standby and migration):** False
- **Database Home ID:** Creates a new database home
- **Database Language:** AMERICAN
- **Database Sizing Template:** odb2
- **Database Storage:** ACFS for version 11.2.0.4 databases. Otherwise, ASM.
- **Database Territory:** AMERICA
- **Database Unique Name:** The user-specified database name and a system-generated suffix, for example, dbtst_phx1cs.
- **PDB Admin Name:** pdbuser (Not applicable for version 11.2.0.4 databases.)

For a list of the database options that you can set, see To launch a DB system.

Using the Console

To launch a DB system

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
3. Click **Launch DB System**.
4. In the **Launch DB System** dialog, enter the following:

   DB System Information

   - **Compartment:** By default, the DB system launches in your current compartment and you can use the network resources in that compartment. Click the **click here** link in the dialog box if you want to enable compartment selection for the DB system, network, and subnet resources.
   - **Display Name:** A friendly, display name for the DB system. The name doesn't

need to be unique. An Oracle Cloud Identifier (OCID) will uniquely identify the DB system.

- **Availability Domain:** The availability domain in which the DB system resides.
- **Shape Type:** The type of shape to use to launch the DB system. The shape type filters the list of available shapes to select from.
- **Shape:** The shape to use to launch the DB system. The shape determines the type of DB system and the resources allocated to the system.

Bare metal shapes

- **BM.DenseIO2.52:** Provides a 1-node DB system (one bare metal server), with up to 52 CPU cores, 768 GB memory, and eight 6.4 TB locally attached NVMe drives (51.2 TB total) to the DB system.
- **BM.DenseIO1.36:** *Limited availability.* Provides a 1-node DB system (one bare metal server), with up to 36 CPU cores, 512 GB memory, and nine 3.2 TB locally attached NVMe drives (28.8 TB total) to the DB system.
  Note: BM.DenseO1.36 is available only to monthly universal credit customers existing on or before November 9th, 2018. This shape is available only in the us-phoenix-1, us-ashburn-1, and eu-frankfurt-1 regions.

Virtual machine shapes

Virtual machine X7 shapes:

- **VM.Standard2.1:** Provides a 1-node DB system with 1 core.
- **VM.Standard2.2:** Provides a 1- or 2-node DB system with 2 cores.
- **VM.Standard2.4:** Provides a 1- or 2-node DB system with 4 cores.
- **VM.Standard2.8:** Provides a 1- or 2-node DB system with 8 cores.

- **VM.Standard2.16:** Provides a 1- or 2-node DB system with 16 cores.
- **VM.Standard2.24:** Provides a 1- or 2-node DB system with 24 cores.

Virtual machine X5 shapes:

- **VM.Standard1.1:** Provides a 1-node DB system with 1 core.
- **VM.Standard1.2:** Provides a 1- or 2-node DB system with 2 cores.
- **VM.Standard1.4:** Provides a 1- or 2-node DB system with 4 cores.
- **VM.Standard1.8:** Provides a 1- or 2-node DB system with 8 cores.
- **VM.Standard1.16:** Provides a 1- or 2-node DB system with 16 cores.



Note

- X5-based shapes availability is limited to monthly universal credit customers existing on or before November 9th, 2018, in the us-phoenix-1, us-ashburn-1, and eu-frankfurt-1 regions.
- VM.Standard1.1 and VM.Standard2.1 shapes cannot be used for 2-node RAC clusters.

- **Cluster Name:** A unique cluster name for a multi-node DB system. The name must begin with a letter and contain only letters (a-z and A-Z), numbers (0-9) and hyphens (-). The cluster name can be no longer than 11 characters and is not case sensitive.
- **Total Node Count:** *Virtual machine DB systems only.* The number of nodes in the DB system. The number depends on the shape you select. You can specify 1 or 2 nodes for virtual machine DB systems, except for VM.Standard2.1 and VM.Standard1.1, which are single-node DB systems.

- **Oracle Database Software Edition:** The database edition supported by the DB system. For bare metal systems, you can mix supported database releases on the DB system to include older database versions, but not editions. The database edition cannot be changed and applies to all the databases in this DB system. Virtual machine systems support only one database.
- **Available Storage Size:** *Virtual machine DB systems only.* The amount of Block Storage you wish to allocate to the virtual machine DB system.
- **Cluster Name:** A unique cluster name for a multi-node DB system. The name must begin with a letter and contain only letters (a-z and A-Z), numbers (0-9) and hyphens (-). The cluster name can be no longer than 11 characters and is not case sensitive.
- **CPU Core Count:** The number of CPU cores for the DB system. Displays only if you select a shape that allows you to configure the number of cores. The text below the field indicates the acceptable values for that shape. For a multi-node DB system, the core count is evenly divided across the nodes.
  Except for virtual machine DB systems, you can increase the CPU cores to accommodate increased demand after you launch the DB system.
- **License Type:** The type of license you want to use for the DB system. Your choice affects metering for billing.
  - **License included** means the cost of the cloud service includes a license for the Database service.
  - **Bring Your Own License (BYOL)** means you are an Oracle Database customer with an Unlimited License Agreement or Non-Unlimited License Agreement and want to use your license with Oracle Cloud Infrastructure. This removes the need for separate on-premises licenses and cloud licenses.
- **SSH Public Key:** The public key portion of the key pair you want to use for SSH access to the DB system. To provide multiple keys, paste each key on a new line. Make sure each key is on a single, continuous line. The length of the combined keys cannot exceed 10,000 characters.

- **Data Storage Percentage:** *For bare metal DB systems only.* The percentage (40% or 80%) assigned to DATA storage (user data and database files). The remaining percentage is assigned to RECO storage (database redo logs, archive logs, and recovery manager backups).
- **Available Storage Size:** *For virtual machine DB systems only.* The amount of block storage to allocate to the virtual machine DB system. Available storage can be scaled up or down as needed after provisioning your DB system.
- **Advanced Options**: *For bare metal DB systems only.*
    - **Disk Redundancy:** *For bare metal systems only.* The type of redundancy configured for the DB system.
        - **Normal** is 2-way mirroring, recommended for test and development systems.
        - **High** is 3-way mirroring, recommended for production systems.

NETWORK INFORMATION

- **Virtual Cloud Network Compartment:** The compartment containing the network in which to launch the DB system.
- **Virtual Cloud Network:** The VCN in which to launch the DB system.
- **Subnet Compartment:** The compartment containing a subnet within the cloud network to attach the DB system to.
- **Client Subnet:** The subnet to which the bare metal or virtual machine DB system should attach.

  For 1- and 2-node RAC DB systems: Do not use a subnet that overlaps with 192.168.16.16/28, which is used by the Oracle Clusterware private interconnect on the database instance. Specifying an overlapping subnet will cause the private interconnect to malfunction.
- **Hostname Prefix:** Your choice of host name for the bare metal or virtual machine DB system. The host name must begin with an alphabetic character, and can contain only alphanumeric characters and hyphens (-). The maximum number

of characters allowed for bare metal and virtual machine DB systems is 16.



Important

The host name must be unique within the subnet. If it is not unique, the DB system will fail to provision.

- **Host Domain Name:** The domain name for the DB system. If the selected subnet uses the Oracle-provided Internet and VCN Resolver for DNS name resolution, this field displays the domain name for the subnet and it can't be changed. Otherwise, you can provide your choice of a domain name. Hyphens (-) are not permitted.
- **Host and Domain URL:** Combines the host and domain names to display the fully qualified domain name (FQDN) for the database. The maximum length is 64 characters.

DATABASE INFORMATION

- **Database Name:** The name for the database. The database name must begin with an alphabetic character and can contain a maximum of eight alphanumeric characters. Special characters are not permitted.
- **Database Version:** The version of the initial database created on the DB system when it is launched. After the DB system is active, you can create additional databases on it. You can mix database versions on the DB system, but not editions.

  If you are launching a DB system with a virtual machine shape, you have option of selecting an older database version. Check **Display all database versions** to include older database versions in the drop-down list of database version choices. See Availability of Older Database Versions for Virtual Machine DB Systems for more information.



Note

When you display all database versions in the drop-down list, the latest database version of each release is represented twice in the list, as follows:

- Once using four numeric segments and the notation "(latest)" . For example: 12.2.0.1 (latest)
- Once using five numeric segments, without the "(latest)" notation. For example: 12.2.0.1.180417

- **PDB Name:** *Not applicable to version 11.2.0.4.* The name of the pluggable database. The PDB name must begin with an alphabetic character, and can contain a maximum of 8 alphanumeric characters. The only special character permitted is the underscore (_).
- **Database Admin Password:**
  A strong password for SYS, SYSTEM, TDE wallet, and PDB Admin. The password must be 9 to 30 characters and contain at least 2 uppercase, 2 lowercase, 2 numeric, and 2 special characters. The special characters must be _, #, or -. The password must not contain the username (SYS, SYSTEM, and so on) or the word "oracle" either in forward or reversed order and regardless of casing.
- **Confirm Database Admin Password:** Re-enter the Database Admin Password you specified.
- **Automatic Backup:** (Optional) If you enable automatic backups, you can choose one of the following preset retention periods: 7 days, 15 days, 30 days, 45 days, 60 days, or 90 days. The default selection is 30 days.

- **Database Workload:**
  Select the workload type that best suits your application.
  - **Online Transactional Processing (OLTP)** configures the database for a transactional workload, with a bias towards high volumes of random data access.
  - **Decision Support System (DSS)** configures the database for a decision support or data warehouse workload, with a bias towards large data scanning operations.
- **Character Set:** The character set for the database. The default is AL32UTF8.
  Advanced Options:
  - **Character Set:** The character set for the database. The default is AL32UTF8.
  - **National Character Set:** The national character set for the database. The default is AL16UTF16.
  - **Fault Domain:** The fault domain(s) in which the DB system resides. You can choose which fault domain to use for your DB system. For 2-node RAC DB systems, you can specify which two fault domains are to be used. Oracle recommends that you place each node of a 2-node RAC DB system in a different fault domain. For more information on fault domains, see Fault Domains.
- **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

5. Click **Launch DB System**.
   The DB system appears in the list with a status of Provisioning. The DB system's icon changes from yellow to green (or red to indicate errors).
6. Wait for the DB system's icon to turn green, with a status of Available, and then click

the highlighted DB system name.
Details about the DB system are displayed.

7. Note the IP addresses; you'll need the private or public IP address, depending on network configuration, to connect to the DB system.

To launch a new DB system from a backup

Before you begin, note the following:

- When you launch a new DB system from a backup, the availability domain will be the same as where the backup is hosted.
- The shape you specify must be the same type as the database from which the backup was taken. For example, if you are using a backup of a 1-node database, then the DB system you select as your target must also be a 1-node DB system.
- The Oracle database software edition you specify must be an equal or greater edition than that of the backed up database.
- If you specify a virtual machine DB system shape, the Available Storage Size will default to the data size of the backup, rounded up to the closest storage size option. However, you can specify a larger storage size.
- If you are creating a database from an automatic backup, you may choose any level 0 weekly backup, or a level 1 incremental backup created after the most recent level 0 backup. For more information on automatic backups, see Automatic Incremental Backups

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. Navigate to the backup you wish to use to create a new database. You can select a backup from a database details page, or select a backup that appears in your compartment's list of standalone backups.

To navigate to a database's list of backups

a. Find the DB system where the database is located, and click the system name to display details about it.
   A list of databases is displayed.
b. Find the database associated with the backup you wish to use, and click its name to display details about it.
   A list of backups is displayed in the default view of the database details. You can also access the list of backups for a database by clicking on **Backups** under **Resources**.

To navigate to the list of standalone backups for your current compartment

a. Click **Standalone Backups** under **Bare Metal, VM, and Exadata**.
b. In the list of standalone backups, find the backup you want to use to create the database.

4. Click the Actions icon (three dots) for the backup you are interested in, and then click **Create Database**.
5. In the **Create Database from Backup** dialog, enter the following:

   DB System Information

   - **DB System Information:** Select **Launch New DB System**.
   - **Display Name:** A friendly, display name for the DB system. The name doesn't need to be unique. An Oracle Cloud Identifier (OCID) will uniquely identify the DB system.
   - **Shape:** The shape to use to launch the DB system. The shape determines the type of DB system and the resources allocated to the system.
     The selected shape must support the same number of nodes as the DB system from which the backup was created.

Bare metal shapes

- **BM.DenseIO2.52:** Provides a 1-node DB system (one bare metal server), with up to 52 CPU cores, 768 GB memory, and eight 6.4 TB locally attached NVMe drives (51.2 TB total) to the DB system.
- **BM.DenseIO1.36:** *Limited availability.* Provides a 1-node DB system (one bare metal server), with up to 36 CPU cores, 512 GB memory, and nine 3.2 TB locally attached NVMe drives (28.8 TB total) to the DB system.
  Note: BM.DenseO1.36 is available only to monthly universal credit customers existing on or before November 9th, 2018. This shape is available only in the us-phoenix-1, us-ashburn-1, and eu-frankfurt-1 regions.

Virtual machine shapes)

Virtual machine X7 shapes:

- **VM.Standard2.1:** Provides a 1-node DB system with 1 core.
- **VM.Standard2.2:** Provides a 1- or 2-node DB system with 2 cores.
- **VM.Standard2.4:** Provides a 1- or 2-node DB system with 4 cores.
- **VM.Standard2.8:** Provides a 1- or 2-node DB system with 8 cores.
- **VM.Standard2.16:** Provides a 1- or 2-node DB system with 16 cores.
- **VM.Standard2.24:** Provides a 1- or 2-node DB system with 24 cores.

Virtual machine X5 shapes:

- **VM.Standard1.1:** Provides a 1-node DB system with 1 core.
- **VM.Standard1.2:** Provides a 1- or 2-node DB system with 2 cores.
- **VM.Standard1.4:** Provides a 1- or 2-node DB system with 4 cores.
- **VM.Standard1.8:** Provides a 1- or 2-node DB system with 8 cores.
- **VM.Standard1.16:** Provides a 1- or 2-node DB system with 16 cores.



Note

- X5-based shapes availability is limited to monthly universal credit customers existing on or before November 9th, 2018, in the us-phoenix-1, us-ashburn-1, and eu-frankfurt-1 regions.
- VM.Standard1.1 and VM.Standard2.1 shapes cannot be used for 2-node RAC clusters.

- **Total Node Count:** *Virtual machine DB systems only.* The number of nodes in the DB system. The number depends on the shape you select. You can specify 1 or 2 nodes for virtual machine DB systems, except for VM.Standard2.1 and VM.Standard1.1, which are single-node DB systems.
- **Oracle Database Software Edition:** The database edition supported by the DB system. For bare metal systems, you can mix supported database releases on the DB system to include older database versions, but not editions. The database edition cannot be changed and applies to all the databases in this DB system. Virtual machine systems support only one database.
- **Available Storage Size:** *Virtual machine DB systems only.* The amount of Block Storage you wish to allocate to the virtual machine DB system.
- **Cluster Name:** A unique cluster name for a multi-node DB system. The name must begin with a letter and contain only letters (a-z and A-Z), numbers (0-9) and hyphens (-). The cluster name can be no longer than 11 characters and is not case sensitive.
- **License Type:** The type of license you want to use for the DB system. Your choice affects metering for billing.

- **License included** means the cost of the cloud service includes a license for the Database service.
- **Bring Your Own License (BYOL)** means you are an Oracle Database customer with an Unlimited License Agreement or Non-Unlimited License Agreement and want to use your license with Oracle Cloud Infrastructure. This removes the need for separate on-premises licenses and cloud licenses.

- **SSH Public Key:** The public key portion of the key pair you want to use for SSH access to the DB system. To provide multiple keys, paste each key on a new line. Make sure each key is on a single, continuous line. The length of the combined keys cannot exceed 10,000 characters.
- **Data Storage Percentage:** *For bare metal DB systems only.* The percentage (40% or 80%) assigned to DATA storage (user data and database files). The remaining percentage is assigned to RECO storage (database redo logs, archive logs, and recovery manager backups).
- **Available Storage Size:** *For virtual machine DB systems only.* The amount of block storage to allocate to the virtual machine DB system. Available storage can be scaled up or down as needed after provisioning your DB system.

  If you are creating a DB system from a backup, the minimum value for available storage is determined by the size of the backup.
- **Advanced Options:**
  - **Disk Redundancy:** *For bare metal systems only.* The type of redundancy configured for the DB system.
    - **Normal** is 2-way mirroring, recommended for test and development systems.
    - **High** is 3-way mirroring, recommended for production systems.
  - **Fault Domain:** The fault domain(s) in which the DB system resides. You can choose which fault domain to use for your DB system. For 2-node RAC DB systems, you can specify which two fault domains are to be used. Oracle

recommends that you place each node of a 2-node RAC DB system in a different fault domain. For more information on fault domains, see Fault Domains.

Network Information

- **Virtual Cloud Network:** The VCN in which to launch the DB system.
- **Client Subnet:** The subnet to which the bare metal or virtual machine DB system should attach.

  For 1- and 2-node RAC DB systems: Do not use a subnet that overlaps with 192.168.16.16/28, which is used by the Oracle Clusterware private interconnect on the database instance. Specifying an overlapping subnet will cause the private interconnect to malfunction.
- **Hostname Prefix:** Your choice of host name for the bare metal or virtual machine DB system. The host name must begin with an alphabetic character, and can contain only alphanumeric characters and hyphens (-). The maximum number of characters allowed for bare metal and virtual machine DB systems is 16.



Important

The host name must be unique within the subnet. If it is not unique, the DB system will fail to provision.

Database Information

- **Database Name:** The name for the database. The database name must begin with an alphabetic character and can contain a maximum of eight alphanumeric characters. Special characters are not permitted.
- **Database Admin Password:**

A strong password for SYS, SYSTEM, TDE wallet, and PDB Admin. The password must be 9 to 30 characters and contain at least 2 uppercase, 2 lowercase, 2 numeric, and 2 special characters. The special characters must be _, #, or -. The password must not contain the username (SYS, SYSTEM, and so on) or the word "oracle" either in forward or reversed order and regardless of casing.

- **Confirm Database Admin Password:** Re-enter the Database Admin Password you specified.
- **Password for Transparent Data Encryption (TDE) Wallet or RMAN Encryption:** Enter either the TDE wallet password or the RMAN encryption password for the backup, whichever is applicable.

6. Click **Create Database**.
7. Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

To check the status of a DB system

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. In the list of DB systems, find the system you're interested in and check its icon. The color of the icon and the text below it indicates the status of the system.
   - **Provisioning:** Yellow icon. Resources are being reserved for the DB system, the system is booting, and the initial database is being created. Provisioning can take several minutes. The system is not ready to use yet.
   - **Available:** Green icon. The DB system was successfully provisioned. A few minutes after the system enters this state, you can SSH to it and begin using it.

- **Terminating:** Gray icon. The DB system is being deleted by the terminate action in the Console or API.
- **Terminated:** Gray icon. The DB system has been deleted and is no longer available.
- **Failed:** Red icon. An error condition prevented the provisioning or continued operation of the DB system.

To view the status of a database node, under Resources, click **Nodes** to see the list of nodes. In addition to the states listed for a DB system, a node's status can be one of the following:

- **Starting:** Yellow icon. The database node is being powered on by the start or reboot action in the Console or API.
- **Stopping:** Yellow icon. The database node is being powered off by the stop or reboot action in the Console or API.
- **Stopped:** Yellow icon. The database node was powered off by the stop action in the Console or API.

You can also check the status of DB systems and database nodes by using the ListDbSystems or ListDbNodes API operations, which return the `lifecycleState` attribute.

To start, stop, or reboot a DB system



Tip

Oracle recommends that you run a Network Time Protocol (NTP) daemon to keep system clocks stable during rebooting. If you need information about an NTP daemon, see Setting Up "NTP (Network Time Protocol) Server" in RHEL/CentOS 7.

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. In the list of DB systems, find the DB system you want to stop or start, and then click its name to display details about it.
4. In the list of nodes, click the Actions icon (three dots) for a node and then click one of the following actions:
   - **Start:** Restarts a stopped node. After the node is restarted, the **Stop** action is enabled.
   - **Stop:** Shuts down the node. After the node is powered off, the **Start** action is enabled.
   - **Reboot:** Shuts down the node, and then restarts it.



Note

- Resource billing differs between bare metal and virtual machine DB systems as follows:
  - **Bare metal DB systems** - The **Stop** state has no effect on the resources you consume. Billing continues for nodes that you stop, and related resources continue to apply against any relevant quotas. You must **Terminate** a DB system to remove its resources from billing and quotas.
  - **Virtual machine DB systems** - Stopping a node stops billing for all OCPUs associated with that node. Billing resumes if you restart the node.
- After you restart or reboot a node, the floating IP address might take several minutes to be updated and display in the Console.

To scale the CPU cores for a bare metal DB system

If a multi-node DB system requires more compute node processing power, you can scale up (burst) the number of enabled CPU cores in the system.



Note

You cannot change the number of CPU cores for a virtual machine DB system.

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. In the list of DB systems, find the system you want to scale and click its highlighted name.
   The system details are displayed.
4. Click **Scale Up/Down** and then change the number in **Total CPU Core Count**. The text below the field indicates the acceptable values, based on the shape used when the DB system was launched.
5. Click **Scale Up/Down DB System**.

To scale up the storage for a virtual machine DB system

If a virtual machine DB system requires more block storage, you can increase the storage at any time without impacting the system.



Note

This procedure does not apply to bare metal DB systems.

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.

A list of DB systems is displayed.

3. In the list of DB systems, find the system you want to scale up and click its highlighted name.
   The system details are displayed.
4. Click **Scale Storage Up**, and then select the new storage size from the drop-down list.
5. Click **Scale Storage Up**.

To terminate a DB system

Terminating a DB system permanently deletes it and any databases running on it.



Note

The database data is local to the DB system and will be lost when the system is terminated. Oracle recommends that you back up any data in the DB system prior to terminating it.

Terminating a DB system removes all automatic incremental backups of all databases in the DB system from Oracle Cloud Infrastructure Object Storage. Full backups remain in Object Storage as standalone backups which you can use to create a new database. See Recovering a Database from Object Storage.

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. For the DB system you want to terminate, click the Actions icon (three dots) and then click **Terminate**.

4. Confirm when prompted. .
   The DB system's icon indicates Terminating.

At this point, you cannot connect to the system and any open connections will be terminated.

To manage your BYOL database licenses

If you want to control the number of database licenses that you run at any given time, you can scale up or down the number of OCPUs on the instance. These additional licenses are metered separately.

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. In the list of DB systems, find the system you want to scale and click its highlighted name.
   The system details are displayed.
4. Click **Scale Up/Down OCPU**, and then change the number.

To manage tags for your DB systems and database resources

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
3. Find the DB system or database resource you're interested in, and click the name.
4. Click the **Tags** tab to view or edit the existing tags. Or click **Apply Tag(s)** to add new ones.

For more information, see Resource Tags.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage DB system components.

DB systems:

- ListDbSystems
- GetDbSystem
- LaunchDbSystem
- TerminateDbSystem

Database homes:

- ListDbHomes
- GetDbHome
- CreateDbHome
- DeleteDbHome

Databases:

- ListDatabases
- GetDatabase

Nodes:

- DbNodeAction: Use this operation to power cycle a node in the DB system.
- ListDbNodes
- GetDbNode

Shapes and database versions:

- ListDbSystemShapes
- ListDbVersions

For the complete list of APIs for the Database service, see Database Service API.

Setting up DNS for a DB System

DNS lets you use host names instead of IP addresses to communicate with a DB system. You can use the *Internet and VCN Resolver* (the DNS capability built into the VCN) as described in DNS in Your Virtual Cloud Network.

Alternatively, you can use your choice of DNS server. You associate the host name and domain name to the public or private IP address of the DB system. You can find the host and domain names and IP addresses for the DB system in the Oracle Cloud Infrastructure Console on the **Database** page.

To associate the host name to the DB system's public or private IP address, contact your DNS administrator and request a custom DNS record for the DB system's IP address. For example, if your domain is example.com and you want to use clouddb1 as the host name, you would request a DNS record that associates clouddb1.example.com to your DB system's IP address.

If you provide the public IP address to your DNS administrator as described above, you should also associate a custom domain name to the DB system's public IP address:

1. Register your domain name through a third-party domain registration vendor, such as register.com.
2. Resolve your domain name to the DB system's public IP address, using the third-party domain registration vendor console. For more information, refer to the third-party domain registration documentation.

Connecting to a DB System

This topic explains how to connect to an active DB system. How you connect depends on the client tool or protocol you use, the purpose of the connection, and how your cloud network is set up. You can find information on various networking scenarios in Overview of Networking,

but for specific recommendations on how you should connect to a database in the cloud, contact your network security administrator.

Prerequisites

This section describes prerequisites you'll need to perform various tasks in this topic.

- To use the Console or the API to get the default administration service connection strings, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in. See Authentication and Authorization for more information on user authorizations for the Oracle Cloud Infrastructure Database service.
- To connect to the database, you'll need the public or private IP address of the DB system. Use the private IP address to connect to the DB system from your on-premises network, or from within the virtual cloud network (VCN). This includes connecting from a host located on-premises connecting through a VPN or FastConnect to your VCN, or from another host in the same VCN. Use the DB System's public IP address to connect to the system from outside the cloud (with no VPN). You can find the IP addresses in the Oracle Cloud Infrastructure Console on the **Database** page.
- For Secure Shell (SSH) access to the DB system, you'll need the full path to the file that contains the private key associated with the public key used when the DB system was launched.

If you have problems connecting, see Troubleshooting Connection Issues.

Database Services and Connection Strings

Database services allow you to control client access to a database instance depending on the functionality needed. For example, you might need to access the database for administration purposes only or you might need to connect an application to the database. Connection strings are specific to a database service.

When you provision a DB system, a default database administration service is automatically created. For 12c and later Oracle Databases, this service is for administrating the database at the CDB level. Because this service provides limited functionality, it is not suitable for connecting an application. Oracle recommends that you create a default application service for the initial database after you create your DB system. For 12c and later Oracle Databases, application services connect at the PDB level. Here are some important functions an application service can provide:

- Workload identification
- Load balancing
- Application continuity and Transaction Guard
- Fast Application Notification
- Resource assignment based on the service name

For details about these and other High Availability capabilities, see Client Failover Best Practices for Highly Available Oracle Databases.

CREATING AN APPLICATION SERVICE

You use the `srvctl` utility to create an application service. Before you can connect to the service, you must start it.

To create an application service for a PDB or an 11g Oracle database

1. Log in to the DB system host as `opc`.
2. Switch to the oracle user, and set your environment to the Oracle Database you want to administer.

```
$ sudo su - oracle
$ . oraenv
ORACLE_SID = [oracle] ? <database_name>
The Oracle base has been set to /u01/app/oracle
```

3. Create the application service for the database. Include the `pdb` option only if you are creating an application service for a PDB.

```
$ srvctl add service
-db <DB_unique_name>
-pdb <PDB_name>
-service <app_service_name>
-role PRIMARY
-notification TRUE
-session_state dynamic
-failovertype transaction
-failovermethod basic
-commit_outcome TRUE
-failoverretry 30
-failoverdelay 10
-replay_init_time 900
-clbgoal SHORT
-rlbgoal SERVICE_TIME
-preferred <rac_node1>,<rac_node2>
-retention 3600
```

Note that the preferred option is required only for multi-node databases to specify the hostname of the node in the RAC.

4. Start the application service.

```
$ srvctl start service -db <DB_unique_name> -s <app_service_name>
```

For more information about services for a PDB, see Managing Services for PDBs.

Database Connection Strings

You must use the appropriate connection string to access a database administration or application service. You can use the Console or the API to get the string for connecting to the default administration service from within a VCN. For 12c and later Oracle Databases, this service is for administrating the database at the CDB level. The string is provided in both the Easy Connect and in the full connect descriptor (long) format. Use the long format for the connection if hostname resolution is not available. You can also use the long format to create an alias in the tnsnames.ora file.

For accessing a database service within the VCN, the connection string for a Real Application Cluster (RAC) DB system uses the Single Client Access Name (SCAN) while the connection string for single instance DB system uses the hostname instead.

The private SCAN name is a Round Robin DNS entry created when you launch a 2-node RAC DB system. The private SCAN name is resolvable only within the VCN. If the client and the database are in the same VCN, the connection mechanism is the same as an on-premises RAC database; all the features provided by VIPs and SCAN VIPs, such as server side load balancing and VIP failover, are available.



Note

If you manually change the DB_UNIQUE_NAME, DB_DOMAIN, or listener port on the DB system, the connection strings you see in the Console or API will not reflect your changes. Ensure that you use the actual values of these parameters when you make a connection.

To get the connection strings for the default administration service

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
3. Find the DB system you're interested in, and click the name.
4. Click **DB Connection**.
5. Click the applicable link to view or copy the connection string.

You can derive the connection strings for other database services by replacing part of the default application service connection string with the applicable values.

To derive the connection string for a PDB administration service or an application service

1. Follow the procedure to get the Easy Connect string for the default administration service. That string should have the following format:

```
<hostname|SCAN>:1521/<DB_unique_name>.<DB_domain>
```

2. Make the appropriate substitution:
   - For the PDB administration service, replace `DB_unique_name` with the PDB name.

     ```
     <hostname|SCAN>:1521/<PDB_name>.<DB_domain>
     ```

   - For an application service, replace `DB_UNIQUE_NAME` with the name of the application service.

     ```
     <hostname|SCAN>:1521/<app_service_name>.<DB_domain>
     ```

Connecting to a Database Service by Using SQL*Net

This section describes how to connect to a database service from a computer that has a SQL*Net client installed. Port 1521 must be open to support the SQL*Net protocol.

Connecting from Within the VCN

For security reasons, Oracle recommends that you connect to your database services from within the VCN. You can use this method whether you are connecting to an administration service or to an application service.

To connect using SQL*Plus, you run the following command using the applicable connection string:

```
sqlplus system/<password>@<connection_string>
```

Consider the following:

- If your system is not using the VCN Resolver, ensure that the DB system's hostname (for single-node systems) or SCAN name (for multi-node systems) can be resolved. See DNS in Your Virtual Cloud Network for information about DNS name resolution.
- For connecting to the administration service of a PDB, ensure that the PDB is open or the service will not be available.
- For connecting to an application service, ensure that the service is started. For Fast Application Notification to work, ensure that port 6200 can be reached. See Client Failover Best Practices for Highly Available Oracle Databases for information about Fast Application Notification.

CONNECTING FROM THE INTERNET

Although Oracle does not recommend connecting to your database from the Internet, you can connect to a database service by using a public IP address if port 1521 is open to the public for ingress.

To use this method, you run the following command using the public IP address instead of the hostname or SCAN in the connection string:

```
sqlplus system/<password>@<public_IP>:1521/<service_name>.<DB_domain>
```

Consider the following:

- SCANs and hostnames are not resolvable on the Internet, therefore load balancing and failover for multi-node DB systems, which rely on these names, cannot work.
- For multi-node DB systems, which normally use SCANs, you must specify the IP address of one of the RAC hosts to access the database.

> **Important**
>
> Do not use this method to connect to the database from within the VCN. Doing so negatively impacts performance because traffic to the database is routed out of the VCN and back in through the public IP address.

EXAMPLE: CONNECTING IN SQL DEVELOPER USING SQL*NET

Prerequisites:

- Ensure that port 1521 is open for the Oracle default listener. (You can do this by checking the DB system's security list.)
- If port 1521 is open only to hosts in the VCN, then you must run your SQL Developer client from a machine that has direct access to the VCN. If you are connecting to the database from the Internet instead, then the public IP address of your computer must be granted access to port 1521 in the security list. (Alternatively, the security list can

grant full access to port 1521, however, this is not recommended for security reasons.) You must use the public IP address of the host because connecting from the Internet does not support SCAN name resolution.

To connect from within the VCN

After the prerequisites are met, start SQL Developer and create a connection by supplying the following connection details:

- **Username:** sys as sysdba
- **Password:** The **Database Admin Password** that was specified in the **Launch DB System** dialog in the Console.
- **Hostname:** The hostname as it appears in the Easy Connect format of the connection string. (See Database Connection Strings for help with getting the connection string and identifying the hostname.)
- **Port:** 1521
- **Service name:** The concatenated name of the service and host domain name, for example, db1_phx1tv.example.com. You can identify this value as the last part of the Easy Connect string, *<service_name>.<DB_domain>*.

To connect from The Internet by using public IP addresses

You can use the node's public IP address to connect to the database if the database is in a VCN that has an internet gateway. However, there are important implications to consider:

- When the client uses the public IP address, the client bypasses the SCAN listener and reaches the node listener, so server side load balancing is not available.
- When the client uses the public IP address, it cannot take advantage of the VIP failover feature. If a node becomes unavailable, new connection attempts to the node will hang until a TCP/IP timeout occurs. You can set client side SQL*Net parameters to limit the TCP/IP timeout.

The following `tnsnames.ora` example shows a connection string that includes the CONNECT_TIMEOUT parameter. This parameter controls the TCP timeout for connecting to a node.

```
test=
  (DESCRIPTION =
    (CONNECT_TIMEOUT=3)
    (ADDRESS_LIST=
      (ADDRESS = (PROTOCOL = TCP)(HOST = <public_IP1>)(PORT = 1521))
      (ADDRESS = (PROTOCOL = TCP)(HOST = <public_IP2>)(PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = <dbservice.subnetname.dbvcn.oraclevcn.com>)
    )
  )
```

For more information about using the CONNECT_TIMEOUT parameter, see Client Failover Best Practices for Highly Available Oracle Databases.

Connecting to a Database with a Public IP by Using SSH Tunneling

You can access the services of DB system databases with public IP addresses by using SSH tunneling. The main advantage of this method is that port 1521 does not need to be opened to the public internet. However, just like accessing the database with a public IP using a SQL*Net client, load balancing and failover for multi-node DB systems cannot work because they rely on SCANs and hostnames.

Oracle SQL Developer and Oracle SQLcL and are two tools that facilitate the use of tunneling for Oracle Database access.

To open a tunnel, and then connect to a database service by using SQLcL, you run commands like the following:

```
SQL> sshtunnel opc@<public_IP> -i <private_key> -L <local_port>:<private_IP>:1521
Using port:22
SSH Tunnel connected
SQL> connect system/<password>@localhost:<local_port>/<service_name>.<DB_domain>
```

See Oracle SQL Developer and Oracle SQLcL for information about these tools.

Connecting to a Database by Using SSH and the Bequeath Protocol

This method allows you to connect to the database without using the network listener. It should be used to connect only for administration purposes.

When connecting to a multi-node DB system, you'll SSH to each individual node in the cluster.

To connect from a UNIX-style system

Use the following SSH command to access the DB system:
`$ ssh -i <private_key> opc@<DB_system_IP_address>`

<private_key> is the full path and name of the file that contains the private key associated with the DB system you want to access.

Use the DB system's private or public IP address depending on your network configuration. For more information, see Prerequisites.

To connect from a Windows system

1. Open `putty.exe`.
2. In the **Category** pane, select **Session** and enter the following fields:
   - **Host Name (or IP address):** `opc@<DB_system_IP_address>`
     Use the DB system's private or public IP address depending on your network configuration. For more information, see Prerequisites.
   - **Connection type:** SSH
   - **Port:** 22
3. In the **Category** pane, expand **Connection**, expand **SSH**, and then click **Auth**, and browse to select your private key.
4. Optionally, return to the **Session** category screen and save this session information for reuse later.
5. Click **Open** to start the session.

To access a database after you connect

1. Log in as opc and then sudo to the grid user.

```
login as: opc

[opc@ed1db01 ~]$ sudo su - grid
```

2. List all the databases on the system.

```
root@ed1db01 ]# srvctl config database -v

cdbm01 /u02/app/oracle/product/12.1.0/dbhome_2 12.1.0.2.0
exadb /u02/app/oracle/product/11.2.0/dbhome_2 11.2.0.4.0
mmdb /u02/app/oracle/product/12.1.0/dbhome_3 12.1.0.2.0
```

3. Connect as the oracle user and get the details about one of the databases by using the `srvctl` command.

```
[root@ed1db01 ~]# su - oracle
[oracle@ed1db01 ~]$ . oraenv
ORACLE_SID = [oracle] ? cdbm01
The Oracle base has been set to /u02/app/oracle
[oracle@ed1db01 ~]$ srvctl config database -d cdbm01
Database unique name: cdbm01 <<== DB unique name
Database name:
Oracle home: /u02/app/oracle/product/12.1.0/dbhome_2
Oracle user: oracle
Spfile: +DATAC1/cdbm01/spfilecdbm01.ora
Password file: +DATAC1/cdbm01/PASSWORD/passwd
Domain: data.customer1.oraclevcn.com
Start options: open
Stop options: immediate
Database role: PRIMARY
Management policy: AUTOMATIC
Server pools:
Disk Groups: DATAC1,RECOC1
Mount point paths:
Services:
Type: RAC
Start concurrency:
```

```
Stop concurrency:
OSDBA group: dba
OSOPER group: racoper
Database instances: cdbm011,cdbm012 <<== SID
Configured nodes: ed1db01,ed1db02
Database is administrator managed
```

4. Set the ORACLE_SID and ORACLE_UNIQUE_NAME using the values from the previous step.

```
[oracle@ed1db01 ~]$ export ORACLE_UNIQUE_NAME=cdbm01
[oracle@ed1db01 ~]$ export ORACLE_SID=cdbm011
[oracle@ed1db01 ~]$ sqlplus / as sysdba

SQL*Plus: Release 12.1.0.2.0 Production on Wed Apr 19 04:10:12 2017

Copyright (c) 1982, 2014, Oracle. All rights reserved.

Connected to:
Oracle Database 12c EE Extreme Perf Release 12.1.0.2.0 - 64bit Production
With the Partitioning, Real Application Clusters, Automatic Storage Management, Oracle Label
Security,
OLAP, Advanced Analytics and Real Application Testing options
```

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the GetDatabase API operation to get the default administration service connection strings.

Troubleshooting Connection Issues

The following issues might occur when connecting to a DB system or database.

ORA-28365: Wallet is Not Open Error

For a 1-node DB system or 2-node RAC DB system, regardless of how you connect to the DB system, *before* you use OS authentication to connect to a database (for example, `sqlplus / as sysdba`) be sure to set the ORACLE_UNQNAME variable. Otherwise, commands that require the TDE wallet will result in the error `ORA-28365: wallet is not open`.

Note that this is not an issue when using a TNS connection because ORACLE_UNQNAME is automatically set in the database CRS resource.

SSH Access Stops Working

If the DB system's root volume becomes full, you might lose the ability to SSH to the system (the SSH command will fail with permission denied errors). Before you copy a large amount of data to the root volume, for example, to migrate a database, use the `dbcli create-dbstorage` command to set up storage on the system's NVMe drives and then copy the database files to that storage. For more information, see Setting Up Storage on the DB System.

What Next?

Before you begin updating your DB system, review the information in Updating a DB System.

For information about setting up an Enterprise Manager console to monitor your databases, see Monitoring a Database.

Updating a DB System



Note

This topic is not applicable to Exadata DB systems. For information on how to update an Exadata DB system, see Updating an Exadata DB System

This topic includes information and instructions on how to update the OS of a bare metal or virtual machine DB system.



Warning

- Review all of the information before you begin updating the system. Updating the operating system through methods not described on this page can cause permanent loss of access.
- Always back up your databases prior to updating your DB system's operating system.

Bash Profile Updates

Do not add interactive commands such as `oraenv`, or commands that might return an error or warning message, to the `.bash_profile` file for the grid or oracle users. Adding such commands can prevent Database service operations from functioning properly.

Essential Firewall Rules

For a 1-node DB system or 2-node RAC DB system, do not remove or modify the following firewall rules in `/etc/sysconfig/iptables`:

- The firewall rules for ports 1521, 7070, and 7060 allow the Database service to manage the DB system. Removing or modifying them can result in the Database Service no longer operating properly.
- The firewall rules for 169.254.0.2:3260 and 169.254.0.3:80 prevent non-root users from escalating privileges and tampering with the system’s boot volume and boot process. Removing or modifying these rules can allow non-root users to modify the system's boot volume.

OS Updates

Before you update the OS, review the following important guidelines and information:

- Back up your DB system's databases prior to attempting an OS update.
- Do not remove packages from a DB system. However, you might have to remove custom RPMs (packages that were installed after the system was provisioned) for the update to complete successfully.



Warning

Do not install NetworkManager on the DB system. Installing this package and rebooting the system results in severe loss of access to the system.

- Oracle recommends that you test any updates thoroughly before updating a production system.
- The image used to launch a DB system is updated regularly with the necessary patches. After you launch a DB system, you are responsible for applying the required OS security updates published through the Oracle public YUM server.
- To apply OS updates, the DB system's VCN must be configured to allow access to the YUM repository. For more information, see Network Setup for DB Systems.

To update the DB system OS



Note

You can update the OS on 2-node RAC virtual machine DB systems in a rolling fashion. Ensure that you stop the clusterware cleanly before starting the update process.

1. Log on to the DB system host as `opc`, and then sudo to the `root` user.

```
login as: opc
[opc@dbsys ~]$ sudo su -
```

2. Identify the host region by running the following command:

```
# curl -s http://169.254.169.254/opc/v1/instance/ |grep region
```

3. With the region you noted from the previous step, determine the region name, and perform the following two steps.
   See Regions and Availability Domains to look up the region name.

   a. Download the repo.

```
# wget https://swiftobjectstorage.<region_
name>.oraclecloud.com/v1/dbaaspatchstore/DBaaSOSPatches/oci_dbaas_ol6repo -O /tmp/oci_
dbaas_ol6repo
```

   This example output assumes the region is us-phoenix-1 (PHX).

```
# wget https://swiftobjectstorage.us-phoenix-
1.oraclecloud.com/v1/dbaaspatchstore/DBaaSOSPatches/oci_dbaas_ol6repo -O /tmp/oci_dbaas_
ol6repo
--2018-03-16 10:40:42-- https://swiftobjectstorage.us-phoenix-
1.oraclecloud.com/v1/dbaaspatchstore/DBaaSOSPatches/oci_dbaas_ol6repo
Resolving swiftobjectstorage.us-phoenix-1.oraclecloud.com... 129.146.13.177,
129.146.13.180, 129.146.12.235, ...
Connecting to swiftobjectstorage.us-phoenix-1.oraclecloud.com|129.146.13.177|:443...
connected.
HTTP request sent, awaiting response... 200 OK
Length: 1394 (1.4K) [binary/octet-stream]
Saving to: `/tmp/oci_dbaas_ol6repo'

100%
[==========================================================================================
==========================================================================================
==================================================>] 1,394 --.-K/s in 0s

2018-03-16 10:40:42 (34.5 MB/s) - `/tmp/oci_dbaas_ol6repo' saved [1394/1394]
```

   b. Download the version lock files.

```
# wget https://swiftobjectstorage.<region_
name>.oraclecloud.com/v1/dbaaspatchstore/DBaaSOSPatches/versionlock.list -O
/tmp/versionlock.list
```

   This example output assumes the region is us-phoenix-1 (PHX).

```
# wget https://swiftobjectstorage.us-phoenix-
1.oraclecloud.com/v1/dbaaspatchstore/DBaaSOSPatches/versionlock.list -O
/tmp/versionlock.list
--2018-03-16 10:41:38-- https://swiftobjectstorage.us-phoenix-
1.oraclecloud.com/v1/dbaaspatchstore/DBaaSOSPatches/versionlock.list
Resolving swiftobjectstorage.us-phoenix-1.oraclecloud.com... 129.146.12.224,
129.146.12.164, 129.146.14.172, ...
Connecting to swiftobjectstorage.us-phoenix-1.oraclecloud.com|129.146.12.224|:443...
connected.
HTTP request sent, awaiting response... 200 OK
Length: 15769 (15K) [binary/octet-stream]
Saving to: `/tmp/versionlock.list'

100%
[==========================================================================================
==========================================================================================
==================================================>] 15,769 --.-K/s in 0.1s

2018-03-16 10:41:39 (123 KB/s) - `/tmp/versionlock.list' saved [15769/15769]
```

4. Enable the repo for your region.
    a. Copy the repo file to the `/etc/yum.repos.d` directory.

    ```
    cp /tmp/oci_dbaas_ol6repo /etc/yum.repos.d/ol6.repo
    ```

    b. Modify the `ol6.repo` file to enable the repo for your region.

    ```
    vi /etc/yum.repos.d/ol6.repo

    [ol6_latest_PHX]
    name=Oracle Linux $releasever Latest ($basearch)
    baseurl=http://yum-phx.oracle.com/repo/OracleLinux/OL6/latest/$basearch/
    gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-oracle
    gpgcheck=1
    enabled=1 <= Enabled.

    [ol6_UEKR4_PHX]
    name=Latest Unbreakable Enterprise Kernel Release 4 for Oracle Linux $releasever
    ($basearch)
    baseurl=http://yum-phx.oracle.com/repo/OracleLinux/OL6/UEKR4/$basearch/
    gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-oracle
    ```

```
gpgcheck=1
enabled=1 <= Enabled.
```

5. Install yum-plugin-versionlock.

```
$ sudo su -
# yum repolist
Loaded plugins: kernel-update-handler, security, ulninfo
ol6_UEKR4
                              | 1.2 kB     00:00
ol6_UEKR4/primary
                              |  29 MB     00:00
ol6_UEKR4
                                        588/588
ol6_latest
                              | 1.4 kB     00:00
ol6_latest/primary
                              |  67 MB     00:00
ol6_latest
                                     39825/39825
repo id                               repo name
                                              status
ol6_UEKR4                             Latest Unbreakable Enterprise Kernel Release 4 for Oracle
Linux 6Server (x86_64)                                    588
ol6_latest                            Oracle Linux 6Server Latest (x86_64)
                                              39825
repolist: 40413
[root@jigsosupg ~]# yum install yum-plugin-versionlock
Loaded plugins: kernel-update-handler, security, ulninfo
Setting up Install Process
Resolving Dependencies
--> Running transaction check
---> Package yum-plugin-versionlock.noarch 0:1.1.30-40.0.1.el6 will be installed
--> Finished Dependency Resolution

Dependencies Resolved


================================================================================================
================================================
 Package                                    Arch                          Version
```

```
              Repository                        Size

================================================================================================================
=====================================================
Installing:
 yum-plugin-versionlock                      noarch                      1.1.30-40.0.1.el6
              ol6_latest                        32 k

Transaction Summary

================================================================================================================
=====================================================
Install       1 Package(s)

Total download size: 32 k
Installed size: 43 k
Is this ok [y/N]: y
Downloading Packages:
yum-plugin-versionlock-1.1.30-40.0.1.el6.noarch.rpm
                              |  32 kB     00:00
warning: rpmts_HdrFromFdno: Header V3 RSA/SHA256 Signature, key ID ec551f03: NOKEY
Retrieving key from file:///etc/pki/rpm-gpg/RPM-GPG-KEY-oracle
Importing GPG key 0xEC551F03:
 Userid : Oracle OSS group (Open Source Software group) <build@oss.oracle.com>
 Package: 6:oraclelinux-release-6Server-8.0.3.x86_64 (@odadom1)
 From   : /etc/pki/rpm-gpg/RPM-GPG-KEY-oracle
Is this ok [y/N]: y
Running rpm_check_debug
Running Transaction Test
Transaction Test Succeeded
Running Transaction
Warning: RPMDB altered outside of yum.
** Found 4 pre-existing rpmdb problem(s), 'yum check' output follows:
oda-hw-mgmt-12.2.0.1.0_LINUX.X64_170614.TR1221-1.x86_64 has missing requires of
/usr/local/bin/perl
oda-hw-mgmt-12.2.0.1.0_LINUX.X64_170614.TR1221-1.x86_64 has missing requires of libnfsodm12.so()
(64bit)
oda-hw-mgmt-12.2.0.1.0_LINUX.X64_170614.TR1221-1.x86_64 has missing requires of perl
(GridDefParams)
oda-hw-mgmt-12.2.0.1.0_LINUX.X64_170614.TR1221-1.x86_64 has missing requires of perl(s_GridSteps)
```

```
  Installing : yum-plugin-versionlock-1.1.30-40.0.1.el6.noarch
                                                        1/1
  Verifying  : yum-plugin-versionlock-1.1.30-40.0.1.el6.noarch
                                                        1/1

Installed:
  yum-plugin-versionlock.noarch 0:1.1.30-40.0.1.el6

Complete!
```



Note

Ignore the `RPMDB` warning messages that refer to `oda-hw-mgmt`.

6. Copy and overwrite the existing version lock file.

```
cp /etc/yum/pluginconf.d/versionlock.list /etc/yum/pluginconf.d/versionlock.list-`date +%Y%m%d`
cp /tmp/versionlock.list /etc/yum/pluginconf.d/versionlock.list
```

   The initial version lock file should be empty. However, it is a good practice to back it up in case it is not and you need to refer to it later.

7. Run the update command.

```
# yum update
Loaded plugins: kernel-update-handler, security, ulninfo, versionlock
Setting up Update Process
Resolving Dependencies
--> Running transaction check
---> Package kernel-uek.x86_64 0:4.1.12-112.14.13.el6uek will be installed
---> Package kernel-uek-firmware.noarch 0:4.1.12-112.14.13.el6uek will be installed
---> Package linux-firmware.noarch 0:20160616-44.git43e96a1e.0.12.el6 will be updated
---> Package linux-firmware.noarch 0:20171128-56.git17e62881.0.2.el6 will be an update
--> Finished Dependency Resolution

Dependencies Resolved
```

```
================================================================================================================
================================================
 Package                              Arch                     Version
                 Repository                   Size

================================================================================================================
================================================
Installing:
 kernel-uek                           x86_64                   4.1.12-112.14.13.el6uek
                 ol6_UEKR4                    51 M
 kernel-uek-firmware                  noarch                   4.1.12-112.14.13.el6uek
                 ol6_UEKR4                   2.4 M
Updating:
 linux-firmware                       noarch                   20171128-56.git17e62881.0.2.el6
                 ol6_UEKR4                    74 M

Transaction Summary

================================================================================================================
================================================
Install       2 Package(s)
Upgrade       1 Package(s)

Total download size: 128 M
Is this ok [y/N]:y
Downloading Packages:
(1/3): kernel-uek-4.1.12-112.14.13.el6uek.x86_64.rpm
                             |  51 MB     00:00
(2/3): kernel-uek-firmware-4.1.12-112.14.13.el6uek.noarch.rpm
                             | 2.4 MB     00:00
(3/3): linux-firmware-20171128-56.git17e62881.0.2.el6.noarch.rpm
                             |  74 MB     00:00
----------------------------------------------------------------------------------------------------------------
------------------------------------------------
Total
                    214 MB/s | 128 MB     00:00
Running rpm_check_debug
Running Transaction Test
Transaction Test Succeeded
Running Transaction
```

```
  Installing : kernel-uek-firmware-4.1.12-112.14.13.el6uek.noarch
                                                    1/4
  Updating   : linux-firmware-20171128-56.git17e62881.0.2.el6.noarch
                                                    2/4
  Installing : kernel-uek-4.1.12-112.14.13.el6uek.x86_64
                                                    3/4
  Cleanup    : linux-firmware-20160616-44.git43e96a1e.0.12.el6.noarch
                                                    4/4
ol6_UEKR4/filelists
                              |  18 MB     00:00
Uploading /boot/vmlinuz-4.1.12-112.14.13.el6uek.x86_64 to http://169.254.0.3/kernel
Uploading /boot/initramfs-4.1.12-112.14.13.el6uek.x86_64.img to http://169.254.0.3/initrd
Uploading /tmp/tmp5HjrRUcmdline to http://169.254.0.3/cmdline


Error activating kernel/initrd/cmdline: 502 - <html>
<head><title>502 Bad Gateway</title></head>
<body bgcolor="white">
<center><h1>502 Bad Gateway</h1></center>
</body>
</html>
```



Note

- Ignore the error activating message that results from running the update.
- An update will occur only if a versionlock file has a valid update available to apply to the DB system.

8. Restart the system.

```
$ sudo su -
# reboot
```

9. Run the following command to validate the update:

```
# uname -r
 4.1.12-112.14.13
```

In this example, then new kernel version is 4.1.12-112.14.13.

For information about applying Oracle database patches to a DB system, see Patching a DB System.

Configuring a DB System

This topic provides information to help you configure your DB System.

Network Time Protocol

Oracle recommends that you run a Network Time Protocol (NTP) daemon on your 1-node DB Systems to keep system clocks stable during rebooting. If you need information about an NTP daemon, see Setting Up "NTP (Network Time Protocol) Server" in RHEL/CentOS 7.

Oracle recommends that you configure NTP on both nodes in a 2-node RAC DB System to synchronize time across the nodes. If you do not configure NTP, then Oracle Clusterware configures and uses the Cluster Time Synchronization Service (CTSS), and the cluster time might be out-of-sync with applications that use NTP for time synchronization.

For information about configuring NTP on a version 12c database, see Setting Network Time Protocol for Cluster Time Synchronization. For a version 11g database, see Network Time Protocol Setting.

Transparent Data Encryption

All user-created tablespaces in a DB System database are encrypted by default, using Transparent Data Encryption (TDE).

- For version 12c databases, if you don't want your tablespaces encrypted, you can set the `ENCRYPT_NEW_TABLESPACES` database initialization parameter to `DDL`.
- On a 1- or 2-node RAC DB System, you can use the dbcli update-tdekey command to update the master encryption key for a database.

- You must create and activate a master encryption key for any PDBs that you create. After creating or plugging in a new PDB on a 1- or 2-node RAC DB System, use the `dbcli update-tdekey` command to create and activate a master encryption key for the PDB. Otherwise, you might encounter the error `ORA-28374: typed master key not found in wallet` when attempting to create tablespaces in the PDB. In a multitenant environment, each PDB has its own master encryption key which is stored in a single keystore used by all containers. For more information, see "Overview of Managing a Multitenant Environment" in the Oracle Database Administrator's Guide.
- For information about encryption on Exadata DB Systems, see Using Tablespace Encryption in Exadata Cloud Service.

For detailed information about database encryption, see the Oracle Database Security White Papers.

Patching a DB System



Note

This topic is not applicable to Exadata DB systems.

This topic explains how to perform patching operations on bare metal and virtual machine DB systems and database homes by using the Console, API, or the database CLI (DBCLI).

Currently, the following patches are available:

Version	DB System Patch	Database Patch
18.0.0.0	April 2019	April 2019, January 2019, October 2018, July 2018, April 2018
12.2.0.1	April 2019	April 2019, January 2019, October 2018, July 2018, April 2018

Version	DB System Patch	Database Patch
12.1.0.2	April 2019	April 2019, January 2019, October 2018, July 2018, April 2018
11.2.0.4	Not applicable	April 2019, January 2019, October 2018, July 2018, April 2018

For information about operating system updates, see OS Updates.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let database admins manage database systems lets the specified group do everything with databases and related Database resources.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for databases, see Details for the Database Service.

About Patching DB Systems

Because patching a system requires a reboot, plan to run the operations at a time when they will have minimal impact on users. To avoid system interruption, consider implementing a high availability strategy such as Oracle Data Guard. For more information, see Using Oracle Data Guard with the Database CLI.

Oracle recommends that you back up your database and test the patch on a test system before you apply the patch. For information about backing up the databases, see Backing Up a Database.

You must patch a DB system before you patch the databases within that system.

Prerequisites

The DB system requires access to the Oracle Cloud Infrastructure Object Storage service, including connectivity to the applicable Swift endpoint for Object Storage. Oracle recommends using a service gateway with the VCN to enable this access. For more information, see these topics:

- Network Setup for DB Systems: For information about setting up your VCN for the DB system, including the service gateway.
- https://cloud.oracle.com/infrastructure/storage/object-storage/faq: For information about the Swift endpoints to use.



Important

In addition to the prerequisites listed, ensure that the following conditions are met to avoid patching failures:

- The `/u01` directory on the database host file system has at least 15 GB of free space for the execution of patching processes.
- The Oracle Clusterware is up and running on the DB system.
- All nodes of the DB system are up and running.

See Patching Failures for details on problems that can result from not following these guidelines.

Using the Console

You can use the Console to view the history of patch operations on a DB system or an individual database, apply patches, and monitor the status of an operation.

Oracle recommends that you use the pre-check action to ensure your DB system or database home has met the requirements for the patch you want to apply.

PERFORMING PATCH OPERATIONS

To perform a patch operation on a DB system

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. Find the DB system on which you want to perform a patch operation, and click its name to display details about it.
4. Under **Resources**, click **Patches**.
5. Review the list of patches.
6. Click the Actions icon (three dots) for the patch you are interested in, and then click one of the following actions:
   - **Pre-check:** Check for any prerequisites to make sure that the patch can be successfully applied.
   - **Apply:** Performs the pre-check, and then applies the patch.
7. Confirm when prompted.
8. In the list of patches, click the patch name to display its patch request and monitor the progress of the patch operation.
   While a patch is being applied, the patch's status displays as **Applying** and the DB system's status displays as **Updating**. If the operation completes successfully, the patch's status changes to **Applied** and the DB system's status changes to **Available**.

To perform a patch operation on a database

Before you perform this procedure, ensure that the latest patch was successfully applied to the DB system.

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.

A list of DB systems is displayed.

3. Find the DB system where the database is located, and click the system name to display details about it.
   A list of databases is displayed.
4. Find the database on which you want to perform the patch operation, and click its name to display details about it.
5. Under **Resources**, click **Patches**.
6. Review the list of patches.
7. Click the Actions icon (three dots) for the patch you are interested in, and then click one of the following actions:
   - **Pre-check:** Check for any prerequisites to make sure that the patch can be successfully applied.
   - **Apply:** Performs the pre-check, and then applies the patch.
8. Confirm when prompted.
9. In the list of patches, click the patch name to display its patch request and monitor the progress of the patch operation.
   While a patch is being applied, the patch's status displays as **Applying** and the database's status displays as **Updating**. If the operation completes successfully, the patch's status changes to **Applied** and the database's status changes to **Available**.

VIEWING PATCH HISTORY

Each patch history entry represents an attempted patch operation and indicates whether the operation was successful or failed. You can retry a failed patch operation. Repeating an operation results in a new patch history entry.

Patch history views in the Console do not show patches that were applied by using command line tools like DBCLI or the Opatch utility.

To view the patch history of a DB system

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. Find the DB system you are interested in, and click the system name to display details about it.
4. Under **Resources**, click **Patch History**.
   The history of patch operations for that DB system is displayed.

To view the patch history of a database

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. Find the DB system where the database is located, and click the system name to display details about it.
   A list of databases is displayed.
4. Find the database you are interested in, and click its name to display details about it.
5. Under **Resources**, click **Patch History**.
   The history of patch operations for that database is displayed.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage patching DB systems and databases.

DB systems:

- ListDbSystemPatches
- ListDbSystemPatchHistoryEntries
- GetDbSystemPatch
- GetDbSystemPatchHistoryEntry
- UpdateDbSystem

Databases:

- ListDbHomePatches
- ListDbHomePatchHistoryEntries
- GetDbHomePatch
- GetDbHomePatchHistoryEntry
- UpdateDbHome

For the complete list of APIs for the Database service, see Database Service API.

Using the Database CLI

This topic explains how to use the command line interface on the DB system to patch a DB system. Patches are available from the Oracle Cloud Infrastructure Object Storage service. You'll use the `dbcli` commands to download and apply patches to some or all of the components in your system.

PREREQUISITES

For connecting to the DB system via SSH, you'll need the path to private key associated with the public key used when the DB system was launched.

You also need the public or private IP address of the DB system. Use the private IP address to connect to the DB system from your on-premises network, or from within the virtual cloud network (VCN). This includes connecting from a host located on-premises connecting through a VPN or FastConnect to your VCN, or from another host in the same VCN. Use the DB System's public IP address to connect to the system from outside the cloud (with no VPN).

You can find the IP addresses in the Oracle Cloud Infrastructure Console on the **Database** page.

To update the CLI with the latest commands

Update the CLI to ensure you have the latest patching commands (older DB systems might not include them).

1. SSH to the DB System.

```
ssh -i <private_key_path> opc@<db_system_ip_address>
```

2. Log in as opc and then sudo to the root user. Use `sudo su -` with a hyphen to invoke the root user's profile, which will set the PATH to the dbcli directory (`/opt/oracle/dcs/bin`).

```
login as: opc

[opc@dbsys ~]$ sudo su -
```

3. Update the CLI by using the cliadm update-dbcli command.

```
[root@dbsys ~]# cliadm update-dbcli
{
  "jobId" : "dc9ce73d-ed71-4473-99cd-9663b9d79bfd",
  "status" : "Created",
  "message" : "Dcs cli will be updated",
  "reports" : [ ],
  "createTimestamp" : "January 18, 2017 10:19:34 AM PST",
  "resourceList" : [ ],
  "description" : "dbcli patching",
  "updatedTime" : "January 18, 2017 10:19:34 AM PST"
}
```

4. Wait for the update job to complete successfully. Check the status of the job by using the dbcli list-jobs command.

```
[root@dbsys ~]# dbcli list-jobs

ID                                       Description                    Created
          Status
---------------------------------------- ------------------------------ --------------------------
```

```
---------- ----------
dc9ce73d-ed71-4473-99cd-9663b9d79bfd     dbcli patching               January 18, 2017 10:19:34
AM PST    Success
```

To check for installed and available patches

1. SSH to the DB System.

   ```
   ssh -i <private_key_path> opc@<db_system_ip_address>
   ```

2. Log in as opc and then sudo to the root user. Use `sudo su -` with a hyphen to invoke the root user's profile, which will set the PATH to the dbcli directory (`/opt/oracle/dcs/bin`).

   ```
   login as: opc

   [opc@dbsys ~]$ sudo su -
   ```

3. Display the installed patch versions by using the dbcli describe-component command. If the **Available Version** column indicates a version number for a component, you should update the component.

   ```
   [root@dbsys ~]# dbcli describe-component
   System Version
   ---------------
   12.1.2.10.0

   Component Name                             Installed Version    Available Version
   ---------------------------------------- -------------------- --------------------
   OAK                                        12.1.2.10.0          up-to-date
   GI                                         12.1.0.2.161018      up-to-date
   ORADB12102_HOME1                           12.1.0.2.160719      12.1.0.2.161018
   ```

4. Display the latest patch versions available in Object Storage by using the dbcli describe-latestpatch command.

   ```
   [root@dbsys ~]# dbcli describe-latestpatch

   componentType   availableVersion
   --------------- --------------------
   gi              12.1.0.2.161018
   ```

```
db              11.2.0.4.161018
db              12.1.0.2.161018
oak             12.1.2.10.0
```

To patch server components

You can patch the Grid Infrastructure (GI) and storage management kit (OAK) server components.

1. SSH to the DB System.

   ```
   ssh -i <private_key_path> opc@<db_system_ip_address>
   ```

2. Log in as opc and then sudo to the root user. Use `sudo su -` with a hyphen to invoke the root user's profile, which will set the PATH to the dbcli directory (`/opt/oracle/dcs/bin`).

   ```
   login as: opc

   [opc@dbsys ~]$ sudo su -
   ```

3. Update the server components by using the dbcli update-server command.

   ```
   [root@dbsys ~]# dbcli update-server
   {
     "jobId" : "9a02d111-e902-4e94-bc6b-9b820ddf6ed8",
     "status" : "Created",
     "reports" : [ ],
     "createTimestamp" : "January 19, 2017 09:37:11 AM PST",
     "resourceList" : [ ],
     "description" : "Server Patching",
     "updatedTime" : "January 19, 2017 09:37:11 AM PST"
   }
   ```

   Note the job ID above.

4. Check the job output by using the dbcli describe-job command with the job ID.

   ```
   [root@dbsys ~]# dbcli describe-job -i 9a02d111-e902-4e94-bc6b-9b820ddf6ed8

   Job details
   ----------------------------------------------------------------
   ```

```
                     ID:  9a02d111-e902-4e94-bc6b-9b820ddf6ed8
            Description:  Server Patching
                 Status:  Running
                Created:  January 19, 2017 9:37:11 AM PST
                Message:

Task Name                                Start Time                          End Time
               Status
---------------------------------------- ----------------------------------- --------------------
-------------- ----------
Create Patching Repository Directories   January 19, 2017 9:37:11 AM PST     January 19, 2017
9:37:11 AM PST     Success
Download latest patch metadata           January 19, 2017 9:37:11 AM PST     January 19, 2017
9:37:11 AM PST     Success
Update System version                    January 19, 2017 9:37:11 AM PST     January 19, 2017
9:37:11 AM PST     Success
Update Patching Repository               January 19, 2017 9:37:11 AM PST     January 19, 2017
9:38:35 AM PST     Success
oda-hw-mgmt upgrade                      January 19, 2017 9:38:35 AM PST     January 19, 2017
9:38:58 AM PST     Success
Opatch updation                          January 19, 2017 9:38:58 AM PST     January 19, 2017
9:38:58 AM PST     Success
Patch conflict check                     January 19, 2017 9:38:58 AM PST     January 19, 2017
9:42:06 AM PST     Success
Apply clusterware patch                  January 19, 2017 9:42:06 AM PST     January 19, 2017
10:02:32 AM PST    Success
Updating GiHome version                  January 19, 2017 10:02:32 AM PST    January 19, 2017
10:02:38 AM PST    Success
```

5. Verify that the server components were updated successfully by using the dbcli describe-component command. The **Available Version** column should indicate `update-to-date`.

To patch database home components

1. SSH to the DB System.

```
ssh -i <private_key_path> opc@<db_system_ip_address>
```

2. Log in as opc and then sudo to the root user. Use `sudo su -` with a hyphen to invoke the

root user's profile, which will set the PATH to the dbcli directory (`/opt/oracle/dcs/bin`).

```
login as: opc

[opc@dbsys ~]$ sudo su -
```

3. Get the ID of the database home by using the dbcli list-dbhomes command.

```
[root@dbsys ~]# dbcli list-dbhomes
ID                                   Name                DB Version Home Location
------------------------------------ ------------------  ---------- ------------------------------
------------
b727bf80-c99e-4846-ac1f-28a81a725df6 OraDB12102_home1   12.1.0.2
/u01/app/orauser/product/12.1.0.2/dbhome_1
```

4. Update the database home components by using the dbcli update-dbhome command and providing the ID from the previous step.

```
[root@dbsys ~]# dbcli update-dbhome -i b727bf80-c99e-4846-ac1f-28a81a725df6
{
  "jobId" : "31b38f67-f993-4f2e-b7eb-5bccda9901ae",
  "status" : "Created",
  "message" : null,
  "reports" : [ ],
  "createTimestamp" : "January 20, 2017 10:08:48 AM PST",
  "resourceList" : [ ],
  "description" : "DB Home Patching: Home Id is 52e2e799-946a-4339-964b-c203dee35328",
  "updatedTime" : "January 20, 2017 10:08:48 AM PST"
}
```

Note the job ID above.

5. Check the job output by using the dbcli describe-job command with the job ID.

```
[root@dbsys ~]# dbcli describe-job -i 31b38f67-f993-4f2e-b7eb-5bccda9901ae

Job details
----------------------------------------------------------------
                     ID:  31b38f67-f993-4f2e-b7eb-5bccda9901ae
            Description:  DB Home Patching: Home Id is b727bf80-c99e-4846-ac1f-28a81a725df6
                 Status:  Success
                Created:  January 20, 2017 10:08:48 AM PST
```

```
                Message:

Task Name                                Start Time                          End Time
                Status
---------------------------------------- ----------------------------------- --------------------
--------------- ----------
Create Patching Repository Directories   January 20, 2017 10:08:49 AM PST    January 20, 2017
10:08:49 AM PST    Success
Download latest patch metadata           January 20, 2017 10:08:49 AM PST    January 20, 2017
10:08:49 AM PST    Success
Update System version                    January 20, 2017 10:08:49 AM PST    January 20, 2017
10:08:49 AM PST    Success
Update Patching Repository               January 20, 2017 10:08:49 AM PST    January 20, 2017
10:08:58 AM PST    Success
Opatch updation                          January 20, 2017 10:08:58 AM PST    January 20, 2017
10:08:58 AM PST    Success
Patch conflict check                     January 20, 2017 10:08:58 AM PST    January 20, 2017
10:12:00 AM PST    Success
db upgrade                               January 20, 2017 10:12:00 AM PST    January 20, 2017
10:22:17 AM PST    Success
```

6. Verify that the database home components were updated successfully by using the dbcli describe-component command. The **Available Version** column should indicate `update-to-date`.

Applying Interim Patches



Note

This topic applies only to database homes in 1-node and 2-node RAC DB systems.

If you are required to apply an interim patch (previously known as a "one-off" patch) to fix a specific defect, follow the procedure in this section. You use the Opatch utility to apply an interim patch to a database home.

In the procedure example, the database home directory is **/u02/app/oracle/product/12.1.0.2/dbhome_1** and the patch number is **26543344**.

To apply an interim patch to a database home

1. Obtain the applicable interim patch from My Oracle Support.
2. Review the information in the patch README.txt file. This file might contain additional and/or custom instructions to follow to apply the patch successfully.
3. Use SCP or SFTP to place the patch on your target database.
4. Shut down each database that is running in the database home.

   ```
   srvctl stop database -db <db name> -stopoption immediate -verbose
   ```

5. Set the Oracle home environment variable to point to the target Oracle home.

   ```
   sudo su - oracle
   export ORACLE_HOME=/u02/app/oracle/product/12.1.0.2/dbhome_1
   ```

6. Change to the directory where you placed the patch, and unzip the patch.

   ```
   cd <work_dir_where_opatch_is stored>
   unzip p26543344_122010_Linux-x86-64.zip
   ```

7. Change to the directory with the unzipped patch, and check for conflicts.

   ```
   cd 26543344
   $ORACLE_HOME/OPatch/opatch prereq CheckConflictAgainstOHWithDetail -ph ./
   ```

8. Apply the patch.

   ```
   $ORACLE_HOME/OPatch/opatch apply
   ```

9. Verify the patch was applied successfully.

   ```
   $ORACLE_HOME/OPatch/opatch lsinventory -detail -oh $ORACLE_HOME
   ```

10. If the database home contains databases, restart them.

    ```
    $ORACLE_HOME/bin/srvctl start database -db <db_name>
    ```

    Otherwise, run the following command as root user.

    ```
    # /u01/app/<db_version>/grid/bin/setasmgidwrap o=/u01/app/oracle/product/<db_version>/dbhome_
    1/bin/oracle
    ```

11. If the readme indicates that the patch has a sqlpatch component, run the datapatch command against each database.
    Before you run datapatch, ensure that all pluggable databases (PDBs) are open. To open a PDB, you can use SQL*Plus to execute `ALTER PLUGGABLE DATABASE <pdb_name> OPEN READ WRITE;` against the PDB.

```
$ORACLE_HOME/OPatch/datapatch
```

Creating a Database



Note

This topic is not applicable to Virtual Machine DB systems.

When you launch a bare metal DB system, an initial database is created in that system. You can create additional databases in that DB system at any time by using the Console or the API. You can create an empty database or reproduce a database by using a backup. Note that if you are creating a database from an automatic backup, you can choose any level 0 weekly backup, or a level 1 incremental backup created after the most recent level 0 backup. For more information on automatic backups, see Automatic Incremental Backups.

The database edition will be the edition of the DB system in which the database is created, and each new database is created in a separate database home.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let database admins manage database systems lets the specified group do everything with databases and related Database resources.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for databases, see Details for the Database Service.

Using the Console

To create a new database in an existing DB system



Note

The database that you create will be the same edition as the initial database in the DB system.

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. In the list of DB systems, find the DB system in which you want to create the database, and then click its name to display details about it.
4. Click **Create Database**.

5. In the **Create Database** dialog, enter the following:
    - **Database Name:** The name for the database. The database name must begin with an alphabetic character and can contain a maximum of eight alphanumeric characters. Special characters are not permitted.
    - **Database Version:** The version of the database. You can mix database versions on the DB system, but not editions.
    - **PDB Name** (Optional) For version 12.1.0.2 and later, you can specify the name of the pluggable database. The PDB name must begin with an alphabetic character, and can contain a maximum of 8 alphanumeric characters. The only special character permitted is the underscore (_).
    - **Admin Password:** A strong password for SYS, SYSTEM, TDE wallet, and PDB Admin. The password must be nine to thirty characters and contain at least two uppercase, two lowercase, two numeric, and two special characters. The special characters must be _, #, or -.
    - **Confirm Admin Password:** Re-enter the database admin password.
    - **Automatic Backup:** *Optional.* If you enable automatic backups, you can choose one of the following preset retention periods: 7 days, 15 days, 30 days, 45 days, 60 days, or 90 days. The default selection is 30 days.
    - **Database Workload:** Select the workload type that best suits your application.
        - **Online Transactional Processing (OLTP)** configures the database for a transactional workload, with a bias towards high volumes of random data access.
        - **Decision Support System (DSS)** configures the database for a decision support or data warehouse workload, with a bias towards large data scanning operations.
6. Optionally, you can specify the character set and/or national character set for this database. Click **Show Advanced Options** to set these parameters. The defaults are AL32UTF8 and AL16UTF16, respectively.

7. Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
8. Click **Create Database**.

When the database creation is complete, the status changes from Provisioning to Available.

To create a database from a backup in an existing DB system

Before you begin, note the following:

- When you create a database from a backup, you can choose a different DB system and compartment. However, the availability domain will be the same as where the source database is hosted.



Tip

You can use the GetBackup API to obtain information about the availability domain of the backup.

- The DB system you specify must support the same type as the system from which the backup was taken. For example, if the backup is from a single-node database, then the target DB system must be a single-node shape.
- The version of the target DB system must be the same or higher than the version of the backup.

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.

3. Navigate to the backup you wish to use to create a new database. You can select a backup from a database details page, or select a backup that appears in your compartment's list of standalone backups.



Tip

If you are creating a database from an automatic backup, you may choose any level 0 weekly backup, or a level 1 incremental backup created after the most recent level 0 backup. For more information on automatic backups, see Automatic Incremental Backups.

To navigate to a database's list of backups

a. Find the DB system where the database is located, and click the system name to display details about it.
   A list of databases is displayed.

b. Find the database associated with the backup you wish to use, and click its name to display details about it.
   A list of backups is displayed in the default view of the database details. You can also access the list of backups for a database by clicking on **Backups** under **Resources**.

To navigate to the list of standalone backups for your current compartment

a. Click **Standalone Backups** under **Bare Metal, VM, and Exadata**.

b. In the list of standalone backups, find the backup you want to use to create the database.

4. Click the Actions icon (three dots) for the backup you are interested in, and then click **Create Database**.
5. In the **Create Database from Backup** dialog, enter the following:
    - **DB System:** The DB system in which you want to create the database. You must have the **Use Existing DB System** radio button selected to see the drop-down list of DB system choices.

      Note

      You cannot create a new database in the same DB system in which the database used to create the backup resides.

    - **Database Name:** The name for the database. The database name must begin with an alphabetic character and can contain a maximum of eight alphanumeric characters. Special characters are not permitted.
    - **Database Admin Password:** A strong password for SYS, SYSTEM, TDE wallet, and PDB Admin. The password must be 9 to 30 characters and contain at least 2 uppercase, 2 lowercase, 2 numeric, and 2 special characters. The special characters must be _, #, or -. The password must not contain the username (SYS, SYSTEM, and so on) or the word "oracle" either in forward or reversed order and regardless of casing.
    - **Confirm Database Admin Password:** Re-enter the Database Admin Password you specified.
    - **Password for Transparent Data Encryption (TDE) Wallet or RMAN Encryption:** Enter either the TDE wallet password or the RMAN encryption password for the backup, whichever is applicable.
6. Click **Create Database from Backup**.
7. Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you

must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

To terminate a database

You'll get the chance to back up the database prior to terminating it. This creates a standalone backup that can be used to create a database later. Oracle recommends that you create this final backup for any production (non-test) database.



Note

Terminating a database removes all automatic incremental backups of the database from Oracle Cloud Infrastructure Object Storage. However, all full backups that were created on demand, including your final backup, will persist as standalone backups.

You cannot terminate a database that is assuming the primary role in a Data Guard association. To terminate it, you can switch it over to the standby role.

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. In the list of DB Systems, find the DB System that contains the database you want to terminate, and then click its name to display details about it.
4. In the list of databases, find the database, click the Actions icon (three dots) and then click **Terminate**.
5. In the confirmation dialog, indicate whether you want to back up the database before terminating it, and type the name of the database to confirm the termination.

6. Click **Terminate Database**.
   The database's status indicates Terminating.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage databases.

Database homes:

- ListDbHomes
- GetDbHome
- CreateDbHome
- DeleteDbHome

Databases:

- ListDatabases
- GetDatabase

For the complete list of APIs for the Database service, see Database Service API.

Monitoring a Database

This topic explains how to set up an:

- Enterprise Manager Express console to monitor a version 12.1.0.2 or later database
- Enterprise Manager Database Control console to monitor a version 11.2.0.4 database

Each console is a web-based database management tool inside the Oracle database. You can use the console to perform basic administrative tasks such as managing user security, memory, and storage, and view performance information.

Required IAM Policy

Some of the procedures below require permission to create or update security lists. For more information about security list policies, see Security Lists.

Monitoring a Database with Enterprise Manager Express

On 1- and 2-node RAC DB Systems, by default, the EM Express console is not enabled on version 18.1.0.0, 12.2.0.1, and 12.1.0.2 databases. You can enable it for an existing database as described below, or you can enable it when you create a database by using the dbcli create-database command with the `-co` parameter.

You must also update the security list and iptables for the DB system as described later in this topic.

When you enable the console, you'll set the port for the console. The procedure below uses port 5500, but each additional console enabled on the same DB system will have a different port.

To enable the EM Express console and determine its port number

1. SSH to the DB system, log in as opc, sudo to the oracle user, and log in to the database as SYS.

   ```
   sudo su - oracle
   . oraenv
    <provide the database SID at the prompt>
   sqlplus / as sysdba
   ```

2. Do one of the following:

   - To enable the console and set its port, use the following command.

     ```
     exec DBMS_XDB_CONFIG.SETHTTPSPORT(<port>);
     ```

     For example:

     ```
     SQL> exec DBMS_XDB_CONFIG.SETHTTPSPORT(5500);

     PL/SQL procedure successfully completed.
     ```

- To determine the port for a previously enabled console, use the following command.

```
select dbms_xdb_config.getHttpsPort() from dual;
```

For example:

```
SQL> select dbms_xdb_config.getHttpsPort() from dual;

DBMS_XDB_CONFIG.GETHTTPSPORT()
------------------------------
                          5500
```

3. Return to the operating system by typing `exit` and then confirm that the listener is listening on the port:

```
lsnrctl status | grep HTTP

(DESCRIPTION=(ADDRESS=(PROTOCOL=tcps)(HOST=xxx.us.oracle.com)(PORT=5500))(Security=(my_wallet_
directory=/u01/app/oracle/admin/prod/xdb_wallet))(Presentation=HTTP)(Session=RAW))
```

4. If you're using a 2-node RAC DB system, see To set the required permissions on a 2-node RAC DB system.
5. Open the console's port as described in Opening Ports on the DB System.
6. Update the security list for the console's port as described in Updating the Security List for the DB System.

To set the required permissions on a 2-node RAC DB system

If you're using a 2-node RAC DB system, you'll need to add read permissions for the asmadmin group on the wallet directory on *both* nodes in the system.

1. SSH to one of the nodes in the DB system, log in as opc, sudo to the grid user.

```
[opc@dbsysHost1 ~]$ sudo su - grid
[grid@dbsysHost1 ~]$ . oraenv
ORACLE_SID = [+ASM1] ?
The Oracle base has been set to /u01/app/grid
```

2. Get the location of the wallet directory, shown in red below in the command output.

```
[grid@dbsysHost1 ~]$ lsnrctl status | grep xdb_wallet

(DESCRIPTION=(ADDRESS=(PROTOCOL=tcps)(HOST=dbsysHost1.sub04061528182.dbsysapril6.oraclevcn.com)
(PORT=5500))(Security=(my_wallet_directory=/u01/app/oracle/admin/dbsys12_phx3wm/xdb_wallet))
(Presentation=HTTP)(Session=RAW))
```

3. Return to the opc user, switch to the oracle user, and change to the wallet directory.

```
[opc@dbsysHost1 ~]$ sudo su - oracle
[oracle@dbsysHost1 ~]$ cd /u01/app/oracle/admin/dbsys12_phx3wm/xdb_wallet
```

4. List the directory contents and note the permissions.

```
[oracle@dbsysHost1 xdb_wallet]$ ls -ltr
total 8
-rw------- 1 oracle asmadmin 3881 Apr  6 16:32 ewallet.p12
-rw------- 1 oracle asmadmin 3926 Apr  6 16:32 cwallet.sso
```

5. Change the permissions:

```
[oracle@dbsysHost1 xdb_wallet]$ chmod 640 /u01/app/oracle/admin/dbsys12_phx3wm/xdb_wallet/*
```

6. Verify that read permissions were added.

```
[oracle@dbsysHost1 xdb_wallet]$ ls -ltr
total 8
-rw-r----- 1 oracle asmadmin 3881 Apr  6 16:32 ewallet.p12
-rw-r----- 1 oracle asmadmin 3926 Apr  6 16:32 cwallet.sso
```

7. **Important!** Repeat the steps above on the other node in the cluster.

To connect to the EM Express console

After you've enabled the console and opened its port in the security list and iptables, you can connect as follows:

1. From a web browser, connect to the console using the following URL format:

```
https://<ip_address>:<port>/em
```

   For example, `https://129.145.0.164:5500/em`

Use the DB system's private or public IP address depending on your network configuration.

Use the private IP address to connect to the DB system from your on-premises network, or from within the virtual cloud network (VCN). This includes connecting from a host located on-premises connecting through a VPN or FastConnect to your VCN, or from another host in the same VCN.

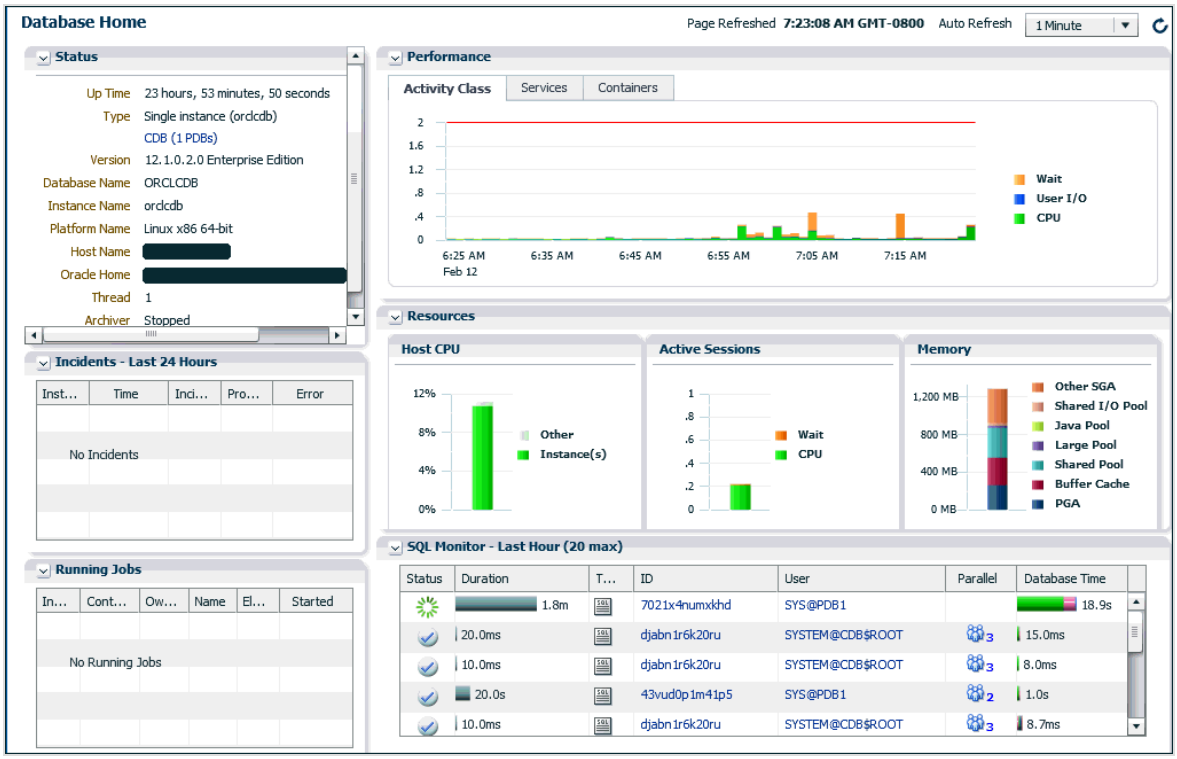
Use the DB System's public IP address to connect to the system from outside the cloud (with no VPN).

You can find the IP addresses in the Oracle Cloud Infrastructure Console on the **Database** page.

2. A login page is displayed and you can log in with any valid database credentials.



The Database Home page is displayed.



To learn more about EM Express, see Introduction to Oracle Enterprise Manager Database Express.

Note

If you're using a 1-node DB system, and you are unable to connect to the EM Express console, see Database Known Issues.

Monitoring a Database with Enterprise Manager Database Control

By default, the Enterprise Manager Database Control console is not enabled on version 11.2.0.4 databases. You can enable the console:

- when you create a database by using the dbcli create-database with the `-co` parameter
- for an existing database as described here.

Port 1158 is the default port used for the first console enabled on the DB system, but each additional console enabled on the DB system will have a different port.



Note

For a version 11.2.0.4 database on a 2-node RAC DB system, see To enable the console for a version 11.2.0.4 database on a multi-node DB system .

To determine the port for the Enterprise Manager Database Control console

1. SSH to the DB system, log in as opc, and sudo to the oracle user.

```
sudo su - oracle
. oraenv
 <provide the database SID at the prompt>
```

2. Use the following command to get the port number.

```
emctl status dbconsole
```

The port is in the URL, as shown in the following example:

```
[oracle@dbsys ~]$ emctl status dbconsole
Oracle Enterprise Manager 11g Database Control Release 11.2.0.4.0
Copyright (c) 1996, 2013 Oracle Corporation.  All rights reserved.
https://dbprod:1158/em/console/aboutApplication
Oracle Enterprise Manager 11g is running.
------------------------------------------------------------------
```

```
Logs are generated in directory /u01/app/oracle/product/11.2.0.4/dbhome_2/dbprod_db11/sysman/log
```

3. Open the console's port as described in Opening Ports on the DB System.
4. Update the security list for the console's port as described in Updating the Security List for the DB System.

To connect to the Enterprise Manager Database Control console

After you've enabled the console and opened its port in the security list and iptables, you can connect as follows:

1. From a web browser, connect to the console using the following URL format:

   ```
   https://<ip_address>:<port>/em
   ```

   For example, `https://129.145.0.164:1158/em`

   Use the DB system's private or public IP address depending on your network configuration.

   Use the private IP address to connect to the DB system from your on-premises network, or from within the virtual cloud network (VCN). This includes connecting from a host located on-premises connecting through a VPN or FastConnect to your VCN, or from another host in the same VCN.

   Use the DB System's public IP address to connect to the system from outside the cloud (with no VPN).

   You can find the IP addresses in the Oracle Cloud Infrastructure Console on the **Database** page.
2. A login page will be displayed and you can log in with any valid database credentials.

To learn more about Enterprise Manager Database Control, see Introduction to Oracle Enterprise Manager Database Control.

To enable the console for a version 11.2.0.4 database on a multi-node DB system

A few extra steps are required to enable the console for a version 11.2.0.4 database on a

multi-node DB system.

Configure SSH Equivalency Between the Two Nodes

You'll create SSH keys on each node and copy the key to the other node, so that each node has the keys for both nodes. The following procedure uses the sample names node1 and node2.

1. SSH to node1, log in as opc, and sudo to the oracle user.

   ```
   sudo su - oracle
   ```

2. Create a directory called .ssh, set its permissions, create an RSA key, and add the public key to the `authorized_keys` file.

   ```
   mkdir .ssh
   chmod 755 .ssh
   ssh-keygen -t rsa
   cat id_rsa.pub > authorized_keys
   ```

3. Repeat the previous steps on the other node in the cluster.
4. On each node, add the `id_rsa.pub` key for the *other* node to the `authorized_keys` file. When you're done, you should see both keys in `authorized_keys` on each node.
5. On node1, create the `known_hosts` file by doing the following:
   - SSH to node1 and reply yes to the authentication prompt.
   - SSH to node2 and reply yes to the authentication prompt.
6. On node2, create the `known_hosts` file by doing the following:
   - SSH to node2 and reply yes to the authentication prompt.
   - SSH to node1 and reply yes to the authentication prompt.
7. On node1, verify that SSH equivalency is now configured by using the following Cluster Verification Utility (CVU) command.

   ```
   cluvfy stage -pre crsinst -n all -verbose
   ```

Configure the Console

1. On node1, create a file called emca.rsp with the following entries.

```
DB_UNIQUE_NAME=<pdb_unique_name>
SERVICE_NAME=<db_unique_name>.<db_domain>
PORT=<scan listener port>
LISTENER_OH=$GI_HOME
SYS_PWD=<admin password>
DBSNMP_PWD=<admin password>
SYSMAN_PWD=<admin password>
CLUSTER_NAME=<cluster name>     <=== to get the cluster name, run: $GI_HOME/bin/cemutlo -n
ASM_OH=$GI_HOME
ASM_SID=+ASM1
ASM_PORT=<asm listener port>
ASM_USER_NAME=ASMSNMP
ASM_USER_PWD=<admin password>
```

2. On node1, run Enterprise Manager Configuration Assistant (EMCA) using the emca.rsp file as input.

```
$ORACLE_HOME/bin/emca -config dbcontrol db -repos create -cluster -silent -respFile  <location of response file above>
```

3. On node2, configure the console so the agent in node1 reports to the console in node1, and the agent in node2 reports to the console in node2.

```
$ORACLE_HOME/bin/emca -reconfig dbcontrol -silent -cluster -EM_NODE <node2 host> -EM_NODE_LIST <node2 host> -DB_UNIQUE_NAME <db_unique_name>
 -SERVICE_NAME <db_unique_name>.<db_domain>
```

4. On each node, verify that console is working properly.

```
$ export ORACLE_UNQNAME=<db_unique_name>

$ emctl status agent
Oracle Enterprise Manager 11g Database Control Release 11.2.0.4.0
Copyright (c) 1996, 2013 Oracle Corporation.  All rights reserved.
---------------------------------------------------------------
Agent Version     : 10.2.0.4.5
OMS Version       : 10.2.0.4.5
Protocol Version  : 10.2.0.4.5
Agent Home        : /u01/app/oracle/product/11.2.0.4/dbhome_x/<host>_<db_unique_name>
Agent binaries    : /u01/app/oracle/product/11.2.0.4/dbhome_x
```

```
Agent Process ID  : 26194
Parent Process ID : 25835
Agent URL         : https://<node host>:1831/emd/main
Repository URL    : https://<node host>:5501/em/upload/
Started at        : 2017-03-15 20:20:34
Started by user   : oracle
Last Reload       : 2017-03-15 20:27:00
Last successful upload                       : 2017-03-15 21:06:36
Total Megabytes of XML files uploaded so far :    22.25
Number of XML files pending upload           :        0                      <=== should be zero
Size of XML files pending upload(MB)         :     0.00
Available disk space on upload filesystem    :    42.75%
Data channel upload directory                : /u01/app/oracle/product/11.2.0.4/dbhome_x/<host>_
<db_unique_name>/sysman/recv
Last successful heartbeat to OMS             : 2017-03-15 21:08:45
---------------------------------------------------------------
```

Update iptables and Security List

1. On each node, edit iptables to open the console's port as described in Opening Ports on the DB System.
2. Update the security list for the console's port as described in Updating the Security List for the DB System.

Opening Ports on the DB System

Open the following ports as needed on the DB system:

- 6200 - For Oracle Notification Service (ONS).
- 5500 - For EM Express. 5500 is the default port, but each additional EM Express console enabled on the DB system will have a different port. If you're not sure which port to open for a particular console, see Monitoring a Database with Enterprise Manager Express.
- 1158 - For Enterprise Manager Database Control. 1158 is the default port, but each additional console enabled on the DB system will have a different port. If you're not sure which port to open for a particular console, see Monitoring a Database with Enterprise Manager Database Control.

For important information about critical firewall rules, see Essential Firewall Rules.

To open ports on the DB system

1. SSH to the DB System.

```
ssh -i <private_key_path> opc@<db_system_ip_address>
```

2. Log in as opc and then sudo to the root user.

```
login as: opc

[opc@dbsys ~]$ sudo su -
```

3. Save a copy of iptables as a backup.

```
[root@dbsys ~]# iptables-save > /tmp/iptables.orig
```

(If necessary, you can restore the original file by using the command `iptables-restore < /tmp/iptables.orig`.)

4. Dynamically add a rule to iptables to allow inbound traffic on the console port, as shown in the following sample. Change the port number and comment as needed.

```
[root@dbsys ~]# iptables -I INPUT 8 -p tcp -m state --state NEW -m tcp --dport 5500 -j ACCEPT -m comment --comment "Required for EM Express."
```

5. Make sure the rule was added.

```
[root@dbsys ~]# service iptables status
```

6. Save the updated file to `/etc/sysconfig/iptables`.

```
[root@dbsys ~]# /sbin/service iptables save
```

The change takes effect immediately and will remain in effect when the node is rebooted.

7. Update the DB system's security list as described in Updating the Security List for the DB System.

Updating the Security List for the DB System

Review the list of ports in Opening Ports on the DB System and for every port you open in iptables, update the security list used for the DB system, or create a new security list.

Note that port 1521 for the Oracle default listener is included in iptables, but should also be added to the security list.

To update an existing security list

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. Locate the DB system in the list.
4. Note the DB system's **Subnet** name and click its **Virtual Cloud Network**.
5. Locate the subnet in the list, and then click its security list under **Security Lists**.
6. Click **Edit All Rules** and add an ingress rule with source type = CIDR, source CIDR=*<source CIDR>*, protocol=TCP, and port=*<port number or port range>*.
   The source CIDR should be the CIDR block that includes the ports you open for the client connection.

For detailed information about creating or updating a security list, see Security Lists.

Backing Up a Database

Backing up your DB System is a key aspect of any Oracle database environment. You can store backups in the cloud or in local storage. Each backup destination has advantages, disadvantages, and requirements that you should consider, as described below.

Object Storage (Recommended)

- Backups are stored in the Oracle Cloud Infrastructure Object Storage.
- Durability: High

- Availability: High
- Back Up and Recovery Rate: Medium
- Advantages: High durability, performance, and availability.

Local Storage

- Backups are stored locally in the Fast Recovery Area of the DB System.
- Durability: Low
- Availability: Medium
- Back Up and Recovery Rate: High
- Advantages: Optimized back up and fast point-in-time recovery.
- Disadvantages: If the DB System becomes unavailable, the backup is also unavailable.

Currently, Oracle Cloud Infrastructure does not provide the ability to attach block storage volumes to a DB System, so you cannot back up to network attached volumes.

For 1- and 2-node RAC DB Systems, see:

- Backing Up to Oracle Cloud Infrastructure Object Storage
- Backing Up to Local Storage Using the Database CLI

Backing Up to Oracle Cloud Infrastructure Object Storage



Note

This topic is not applicable to Exadata DB systems. For Exadata DB systems, see Backing Up an Exadata Database.

This topic explains how to work with backups managed by Oracle Cloud Infrastructure. You do this by using the Console or the API. (For unmanaged backups, you can use `RMAN` or `dbcli`,

and you must create and manage your own Object Storage buckets for backups. See Backing Up to Object Storage Using RMAN.)



Warning

If you previously used `RMAN` or `dbcli` to configure backups and then you switch to using the Console or the API for backups, a new backup configuration is created and associated with your database. This means that you can no longer rely on your previously configured unmanaged backups to work.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

If you're new to policies, see Getting Started with Policies and Common Policies.

Prerequisites

The DB system requires access to the Oracle Cloud Infrastructure Object Storage service, including connectivity to the applicable Swift endpoint for Object Storage. Oracle recommends using a service gateway with the VCN to enable this access. For more information, see these topics:

- Network Setup for DB Systems: For information about setting up your VCN for the DB system, including the service gateway.
- https://cloud.oracle.com/infrastructure/storage/object-storage/faq: For information about the Swift endpoints to use.



Important

In addition to the prerequisites listed, ensure that the following conditions are met to avoid backup failures:

- The database's archiving mode is set to `ARCHIVELOG` (the default).
- The `/u01` directory on the database host file system has sufficient free space for the execution of backup processes.
- The .bash_profile file for the oracle user does not include any interactive commands (such as `oraenv` or one that could generate an error or warning message).
- (For automatic backups) No changes were made to the default `WALLET_LOCATION` entry in the sqlnet.ora file.
- No changes were made to `RMAN` backup settings by using standard `RMAN` commands.

See Backup Failures for details on problems that can result from not following these guidelines.

USING THE CONSOLE

You can use the Console to enable automatic incremental backups, create full backups on demand, and view the list of managed backups for a database. The Console also allows you to delete full backups.



Note

The list of backups you see in the Console does not include any unmanaged backups (backups created directly by using `RMAN` or `dbcli`).

All backups are encrypted with the same master key used for Transparent Data Encryption (TDE) wallet encryption.

The database and DB system must be in an "Available" state for a backup operation to run successfully. Oracle recommends that you avoid performing actions that could interfere with availability (such as patching and Data Guard operations) while a backup operation is in progress. If an automatic backup operation fails, the Database service retries the operation during the next day's backup window. If an on-demand full backup fails, you can try the operation again when the DB system and database availability are restored.

Automatic Incremental Backups

When you enable the Automatic Backup feature, the service creates daily incremental backups of the database to Object Storage. The first backup created is a level 0 backup. Then, level 1 backups are created every day until the next weekend. Every weekend, the cycle repeats, starting with a new level 0 backup. The automatic backup process can run at any time within the daily backup window (between midnight and 6:00 AM in the time zone of the DB system's region). See note for backup window time zone information. If you choose to enable automatic backups, you can choose one of the following preset retention periods: 7 days, 15 days, 30 days, 45 days, or 60 days. The system automatically deletes your incremental backups at the end of your chosen retention period.



Note

- **Backup Window Time Zone** - Automatic backups enabled for the first time after November 20, 2018 on any database will run between midnight and 6:00 AM in the time zone of the DB system's region. If you have enabled automatic backups on a database before this date, the backup window for the database will continue to be between midnight and 6:00 AM **UTC**. You can create a My Oracle Support service request to have your automatic backups run in a backup window of your choice.
- **Data Guard** - You can enable the Automatic Backup feature on a database with the standby role in a Data Guard association. However, automatic backups for that database will not be created until it assumes the primary role.
- **Retention Period Changes** - If you shorten your database's automatic backup retention period in the future, existing backups falling outside the updated retention period are deleted by the system.
- **Object Storage Costs** - Automatic backups incur Object Storage usage costs.

On-Demand Full Backups

You can create a full backup of your database at any time unless your database is assuming the standby role in a Data Guard association.

Standalone Backups

When you terminate a DB system or a database, all of its resources are deleted, along with any automatic backups. Full backups remain in Object Storage as standalone backups. You can use a standalone backup to create a new database.

To navigate to the list of standalone backups for your current compartment

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Click **Standalone Backups** under **Bare Metal, VM, and Exadata**.

To configure automatic backups for a database

When you launch a DB system, you can optionally enable automatic backups for the initial database. Use this procedure to configure or disable automatic backups after the database is created.

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. Find the DB system where the database is located, and click the system name to display details about it.
   A list of databases is displayed.
4. Find the database for which you want to enable or disable automatic backups, and click its name to display database details. The details indicate whether automatic backups are enabled. When backups are enabled, the details also indicate the chosen backup retention period .
5. Click **Configure Automatic Backups**.
6. In the **Configure Automatic Backups** dialog, check or uncheck **Enable Automatic Backup**, as applicable. If you are enabling automatic backups, you can choose one of

the following preset retention periods: 7 days, 15 days, 30 days, 45 days, or 60 days. The default selection is 30 days.

7. Click **Save Changes**.

To create an on-demand full backup of a database

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. Find the DB system where the database is located, and click the system name to display details about it.
   A list of databases is displayed.
4. Find the database for which you want to create an on-demand full backup and click its name to display database details.
5. Under **Resources**, click **Backups**.
   A list of backups is displayed.
6. Click **Create Backup**.

To delete full backups from Object Storage



Note

You cannot explicitly delete automatic backups. Unless you terminate the database, automatic backups remain in Object Storage for 30 days, after which time they are automatically deleted.

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. Find the DB system where the database is located and click the DB system name to display details.
   A list of databases is displayed.
4. Find the database you are interested in and click its name to display database details.
5. Under **Resources**, click **Backups**.
   A list of backups is displayed.
6. Click the Actions icon (three dots) for the backup you are interested in, and then click **Delete**.
7. Confirm when prompted.

USING THE API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage database backups:

- ListBackups
- GetBackup
- CreateBackup
- DeleteBackup
- UpdateDatabase - To enable and disable automatic backups.

For the complete list of APIs for the Database service, see Database Service API.

WHAT'S NEXT?

See Recovering a Database from Object Storage.

Backing Up to Local Storage Using the Database CLI



Note

This topic is not applicable to virtual machine DB systems because they have no local storage. For Exadata DB systems, see Backing Up an Exadata Database.

This topic explains how to back up to the local Fast Recovery Area on a bare metal DB system by using the database CLI (dbcli). Some sample dbcli commands are provided below. For complete command syntax, see the Oracle Database CLI Reference.



Note

Backing up to local storage is fast and provides for fast point-in-time recovery, however, if the DB system becomes unavailable, the backup also becomes unavailable. For information about more durable backup destinations, see Backing Up a Database.

Backing Up the Database to Local Storage

You'll use the dbcli commands to create a backup configuration, associate the backup configuration with the database, initiate the backup operation, and then review the backup job.

1. SSH to the DB System.

   ```
   ssh -i <private_key_path> opc@<db_system_ip_address>
   ```

2. Log in as opc and then sudo to the root user. Use `sudo su -` with a hyphen to invoke the root user's profile, which will set the PATH to the dbcli directory (`/opt/oracle/dcs/bin`).

```
login as: opc

[opc@dbsys ~]$ sudo su -
```

3. Create a backup configuration by using the dbcli update-backupconfig command and specify local disk storage as the backup destination.

   The following example creates a backup configuration named prodbackup and specifies a disk backup destination and a disk recovery window of 5 (backups and archived redo logs will be maintained in local storage for 5 days).

```
[root@dbsys ~]# dbcli create-backupconfig --name prodbackup --backupdestination disk  --
recoverywindow 5
{
  "jobId" : "e7050756-0d83-48ce-9336-86592be59827",
  "status" : "Success",
  "message" : null,
  "reports" : [ {
    "taskId" : "TaskParallel_471",
    "taskName" : "persisting backup config metadata",
    "taskResult" : "Success",
    "startTime" : 1467774813141,
    "endTime" : 1467774813207,
    "status" : "Success",
    "taskDescription" : null,
    "parentTaskId" : "TaskSequential_467",
    "jobId" : "e7050756-0d83-48ce-9336-86592be59827",
    "reportLevel" : "Info",
    "updatedTime" : 1467774813207
  } ],
  "createTimestamp" : 1467774781851,
  "description" : "create backup config:prodbackup",
  "updatedTime" : 1467774813236
}
```

   The example above uses full parameter names for demonstration purposes, but you can abbreviate the parameters like this:

```
dbcli create-backupconfig -n prodbackup -d disk  -w 5
```

4. Get the ID of the database you want to back up by using the dbcli list-databases command.

```
[root@dbsys ~]# dbcli list-databases

ID                                       DB Name    DB Version CDB        Class    Shape
Storage    Status
---------------------------------------- ---------- ---------- ---------- -------- -------- -----
----- ----------
71ec8335-113a-46e3-b81f-235f4d1b6fde     prod       12.1.0.2   true       OLTP     odb1     ACFS
     Configured
```

5. Get the ID of the backup configuration by using the dbcli list-backupconfigs command.

```
[root@dbbackup backup]# /opt/oracle/dcs/bin/dbcli list-backupconfigs
ID                                       Name                 DiskRecoveryWindow
BackupDestination createTime
---------------------------------------- -------------------- ----- ------ ----------------------
-------------

78a2a5f0-72b1-448f-bd86-cf41b30b64ee     prodbackup           5     Disk   July 6, 2016 3:13:01
AM UTC
```

6. Associate the backup configuration ID with the database ID by using the dbcli update-database command.

```
[root@dbsys ~]# dbcli update-database  --backupconfigid 78a2a5f0-72b1-448f-bd86-cf41b30b64ee --
dbid 71ec8335-113a-46e3-b81f-235f4d1b6fde
{
  "jobId" : "2b104028-a0a4-4855-b32a-b97a37f5f9c5",
  "status" : "Created",
  "message" : null,
  "reports" : [ ],
  "createTimestamp" : 1467775842977,
  "description" : "update database id:71ec8335-113a-46e3-b81f-235f4d1b6fde",
  "updatedTime" : 1467775842978
}
```

   You can view details about the update job by using the dbcli describe-job command and specifying the job ID from the `dbcli update-database` command output, for example:

```
dbcli describe-job --jobid 2b104028-a0a4-4855-b32a-b97a37f5f9c5
```

7. Initiate the database backup by using the dbcli create-backup command. The backup operation is performed immediately.

The following example creates a backup of the specified database.

```
[root@dbsys ~]# dbcli create-backup --dbid 71ec8335-113a-46e3-b81f-235f4d1b6fde
{
    "createTimestamp": 1467792576854,
    "description": "Backup service creation with db name: prod",
    "jobId": "d6c9edaa-fc80-40a9-bcdd-056430cdc56c",
    "message": null,
    "reports": [],
    "status": "Created",
    "updatedTime": 1467792576855
}
```

Or you can abbreviate the command parameters like this:

```
dbcli create-backup -i 71ec8335-113a-46e3-b81f-235f4d1b6fde
```

You can view details about the back up job by using the dbcli describe-job command and specifying the job ID from the `dbcli create-backup` command output, for example:

```
dbcli describe-job --jobid d6c9edaa-fc80-40a9-bcdd-056430cdc56c
```

8. **Important!** Manually back up any TDE password-based wallets to your choice of a safe location, preferably not on the DB system. The wallets are required to restore the backup to a new host.
9. Optionally, you can review the backup report. Use the Oracle Database CLI Reference command to create a report, then use Oracle Database CLI Reference to get the report ID, and then use Oracle Database CLI Reference with the report ID to get the report location, as shown in the following example.

```
[root@dbsys ~]# dbcli create-backupreport --dbid 71ec8335-113a-46e3-b81f-235f4d1b6fde --
reporttype summary

{
       "jobId" : "65ce79fe-4ef4-4d7d-8020-e56a5390026d",
       "status" : "Created",
       "message" : null,
       "reports" : [ ],
       "createTimestamp" : "July 6, 2016 23:06:11 PM UTC",
       "description" : "Creating a report for database 71ec8335-113a-46e3-b81f-235f4d1b6fde",
       "updatedTime" : "July 6, 2016 23:06:11 PM UTC"
```

```
}

[root@dbsys ~]# dbcli list-backupreports

ID                                       Name                 ReportType DbId
                createTime                          updatedTime
---------------------------------------- -------------------- ---------- ------------------------
---------------- ---------------------------------- ----------------------------------
c0e0a16a-485f-4176-ab73-5b30ccf5c560                          summary    71ec8335-113a-46e3-b81f-
235f4d1b6fde     July 6, 2016 11:04:05 PM UTC        July 6, 2016 11:04:17 PM UTC


[root@dbsys ~]# dbcli describe-backupreport --id c0e0a16a-485f-4176-ab73-5b30ccf5c560

Backup Report details
----------------------------------------------------------------
                     ID: ed67a2cf-fe63-4755-a5d6-7eda5b669837
                   Name:
            Report Type: summary
               Location:
/opt/oracle/dcs/log/LrbvevghazqOGpbatvbpMRZJeVCzaW/rman/bkup/hhUiFnBz/rman_list_backup_
summary/2016-07-06/rman_list_backup_summary_2016-07-06_09-49-20.0832.log
            Database ID: 71ec8335-113a-46e3-b81f-235f4d1b6fde
            CreatedTime: July 6, 2016 3:49:12 AM UTC
            UpdatedTime: July 6, 2016 3:49:24 AM UTC
```

After the backup command completes, the database backup files are available in the Fast Recovery Area on the DB system.

WHAT'S NEXT?

See Recovering a Database from a CLI Backup.

Recovering a Database

For information on restoring a database on a bare metal or virtual machine DB system, see the following topics:

- Recovering a Database from Object Storage
- Recovering a Database from a CLI Backup

- Recovering a Database from the Oracle Cloud Infrastructure Classic Object Store

Recovering a Database from Object Storage



Note

This topic is not applicable to Exadata DB systems.

This topic explains how to recover a database from a backup stored in Object Storage. The service is a secure, scalable, on-demand storage solution in Oracle Cloud Infrastructure. For information on using Object Storage as a backup destination, see Backing Up to Oracle Cloud Infrastructure Object Storage.

You can recover a database using the Console, API, or by using RMAN.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

If you're new to policies, see Getting Started with Policies and Common Policies.

PREREQUISITES

The DB system requires access to the Oracle Cloud Infrastructure Object Storage service, including connectivity to the applicable Swift endpoint for Object Storage. Oracle recommends using a service gateway with the VCN to enable this access. For more information, see these topics:

- Network Setup for DB Systems: For information about setting up your VCN for the DB system, including the service gateway.
- https://cloud.oracle.com/infrastructure/storage/object-storage/faq: For information about the Swift endpoints to use.

USING THE CONSOLE

You can use the Console to restore the database from a backup in the Object Storage that was created by using the Console or the API. You can restore to the last known good state of the database, or you can specify a point in time or an existing System Change Number (SCN). You can also create a new database by using a standalone backup.



Note

The list of backups you see in the Console does not include any unmanaged backups (backups created directly by using `RMAN` or `dbcli`).

Restoring a database with Data Guard enabled is not supported. You must first remove the Data Guard association by terminating the standby database before you can restore the database.

RESTORING AN EXISTING DATABASE

To restore a database

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.

2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. Find the DB system where the database is located, and click the system name to display details about it.
   A list of databases is displayed.
4. Find the database you want to restore, and click its name to display details about it.
   A list of backups is displayed in the default view of the database details. You can also access the list of backups for a database by clicking on **Backups** under **Resources**.
5. Click the Actions icon (three dots) for the backup you are interested in, and then click **Restore**.
6. Select one of the following options, and click **Restore Database**:
   - **Restore to the latest:** Restores the database to the last known good state with the least possible data loss.
   - **Restore to the timestamp:** Restores the database to the timestamp specified.
   - **Restore to SCN:** Restores the database using the SCN specified. This SCN must be valid.

   Tip

   You can determine the SCN number to use either by accessing and querying your database host, or by accessing any online or archived logs.
7. Confirm when prompted.
   If the restore operation fails, the database will be in a "Restore Failed" state. You can try restoring again using a different restore option. However, Oracle recommends that you review the `RMAN` logs on the host and fix any issues before reattempting to restore the database.

To restore a database using a specific backup from Object Storage

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose your **Compartment**.
   A list of DB systems is displayed.
3. Find the DB system where the database is located, and click the system name to display details about it.
   A list of databases is displayed.
4. Find the database you want to restore, and click its name to display details about it.
5. Under **Resources**, click **Backups**.
   A list of backups is displayed.
6. Click the Actions icon (three dots) for the backup you are interested in, and then click **Restore**.
7. Confirm when prompted.

Creating a New Database from a Backup

You can use a backup to create a database in an existing DB system or to launch a new DB system. See the following procedures for more information:

- To create a database from a backup in an existing DB system
- To launch a new DB system from a backup

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to recover a database:

- ListBackups
- GetBackup

- RestoreDatabase
- CreateDbHome - For creating a DB system database from a standalone backup.

For the complete list of APIs for the Database service, see Database Service API.

Using an RMAN Backup

This topic explains how to recover a Recovery Manager (RMAN) backup stored in Object Storage.

Prerequisites

You'll need the following:

- A new DB system to restore the database to (see assumptions below). For more information, see Managing Bare Metal and Virtual Machine DB Systems.
- The Oracle Database Cloud Backup Module must be installed on the DB system. For more information, see Installing the Backup Module on the DB System.

Assumptions

The procedures below assume the following:

- A new DB system has been created to host the restored database and no other database exists on the new DB system. It is possible to restore to a DB system that has existing databases, but that is beyond the scope of this topic.
- The original database is lost and all that remains is the latest RMAN backup. For virtual machine DB systems, the procedure assumes the DB system (inclusive of the database) no longer exists.



Warning

Any data not included in the most recent backup will be lost.

- The Oracle Wallet and/or encryption keys used by the original database at the time of the last backup is available.
- The RMAN backup contains a copy of the control file and spfile as of the most recent backup as well as all of the datafile and archivelog backups needed to perform a complete database recovery.
- An RMAN catalog will not be used during the restore.

SETTING UP STORAGE ON THE DB SYSTEM

1. SSH to the DB System.

   ```
   ssh -i <private_key_path> opc@<db_system_ip_address>
   ```

2. Log in as opc and then sudo to the root user. Use `sudo su -` with a hyphen to invoke the root user's profile, which will set the PATH to the dbcli directory (`/opt/oracle/dcs/bin`).

   ```
   login as: opc

   [opc@dbsys ~]$ sudo su -
   ```

3. You can use an existing empty database home or create a new one for the restore. Use the applicable commands to help you complete this step.

   If you will be using an existing database home:

   - Use the dbcli list-dbhomes command to list the database homes.

     ```
     [root@dbsys ~]# dbcli list-dbhomes
     ID                                       Name                 DB Version Home Location
     ---------------------------------------- -------------------- ---------- -----------------
     ----------------------------
     2e743050-b41d-4283-988f-f33d7b082bda     OraDB12102_home1     12.1.0.2
     /u01/app/oracle/product/12.1.0.2/dbhome_1
     ```

   - Use the dbcli list-databases command to ensure the database home is not associated with any database.

   If necessary, use the dbcli create-dbhome command to create a database home for the restore.

4. Use the dbcli create-dbstorage to set up directories for DATA, RECO, and REDO storage. The following example creates 10GB of ACFS storage for the rectest database.

```
[root@dbsys ~]# dbcli create-dbstorage --dbname rectest --dataSize 10 --dbstorage ACFS
```



Note

When restoring a version 11.2 database, ACFS storage must be specified.

PERFORMING THE DATABASE RESTORE AND RECOVERY

1. SSH to the DB system, log in as opc, and then become the oracle user.

```
sudo su - oracle
```

2. Create an entry in `/etc/oratab` for the database. Use the same SID as the original database.

```
db1:/u01/app/oracle/product/12.1.0.2/dbhome_1:N
```

3. Set the ORACLE_HOME and ORACLE_SID environment variables using the oraenv utility.

```
. oraenv
```

4. Obtain the DBID of the original database. This can be obtained from the file name of the `controlfile` autobackup on the backup media. The file name will include a string that contains the DBID. The typical format of the string is `c-DDDDDDDDDDDD-YYYYMMDD-NN` where `DDDDDDDDDDDD` is the DBID, `YYYYMMDD` is the date the backup was created, and `NN` is a sequence number to make the file name unique. The DBID in the following examples is 1508405000. Your DBID will be different.

   Use the following curl syntax to perform a general query of Object Storage. The parameters in red are the same parameters you specified when installing the backup module as described in Installing the Backup Module on the DB System.

```
curl -u '<user_ID>.com:<auth_token>' -v https://swiftobjectstorage.<region_
name>.oraclecloud.com/v1/<tenant_name>
```

See Regions and Availability Domains to look up the region name.
For example:

```
curl -u 'djones@mycompany.com:1cnk!d0++ptETd&C;tHR' -v https://swiftobjectstorage.<region_name>.oraclecloud.com/v1/mycompany
```

To get the DBID from the control file name, use the following syntax:

```
curl -u '<user_id>.com:<auth_token>' -v https://swiftobjectstorage.<region_name>.oraclecloud.com/v1/<tenant_name>/<bucket_name>?prefix=sbt_catalog/c-
```

For example:

```
curl -u 'djones@mycompany.com:1cnk!d0++ptETd&C;tHR' -v https://swiftobjectstorage.<region_name>.oraclecloud.com/v1/mycompany/dbbackups/?prefix=sbt_catalog/c-
```

In the sample output below, 1508405000 is the DBID.

```
{
    "bytes": 1732,
    "content_type": "binary/octet-stream",
    "hash": "f1b61f08892734ed7af4f1ddaabae317",
    "last_modified": "2016-08-11T20:28:34.438000",
    "name": "sbt_catalog/c-1508405000-20160811-00/metadata.xml"
}
```

5. Run RMAN and connect to the target database. There is no need to create a `pfile` or `spfile` or use a backup `controlfile`. These will be restored in the following steps. Note that the target database is `(not started)`. This is normal and expected at this point.

```
rman target /

Recovery Manager: Release 12.1.0.2.0 - Production on Wed Jun 22 18:36:40 2016

Copyright (c) 1982, 2014, Oracle and/or its affiliates.  All rights reserved.

connected to target database (not started)
```

6. Set the DBID using the value obtained above.

```
RMAN> set dbid 1508405000;

executing command: SET DBID
```

7. Run the `STARTUP NOMOUNT` command. If the server parameter file is not available, RMAN attempts to start the instance with a dummy server parameter file. The ORA-01078 and LRM-00109 errors are normal and can be ignored.

```
RMAN> STARTUP NOMOUNT

startup failed: ORA-01078: failure in processing system parameters
LRM-00109: could not open parameter file '/u01/app/oracle/product/12.1.0.2/dbhome_
1/dbs/initdb1.ora'

starting Oracle instance without parameter file for retrieval of spfile
Oracle instance started

Total System Global Area    2147483648 bytes

Fixed Size                     2944952 bytes
Variable Size                847249480 bytes
Database Buffers            1254096896 bytes
Redo Buffers                  43192320 bytes
```

8. Restore the server parameter file from autobackup.
   The SBT_LIBRARY is the same library specified with the `-libDir` parameter when the Backup Module was installed, for example `/home/oracle/lib/`.
   The OPC_PFILE is the same file specified with the `-configfile` parameter when the Backup Module was installed, for example `/home/oracle/config`.

```
set controlfile autobackup format for device type sbt to '%F';
run {
  allocate channel c1 device type sbt PARMS  'SBT_LIBRARY=/home/oracle/lib/libopc.so, SBT_PARMS=
(OPC_PFILE=/home/oracle/config)';
  restore spfile from autobackup;
}
```

9. Create the directory for audit_file_dest. The default is `/u01/app/oracle/admin/$ORACLE_SID/adump`. You can see the setting used by the original database by searching the spfile for the string, `audit_file_dest`.

```
strings ${ORACLE_HOME}/dbs/spfile${ORACLE_SID}.ora | grep audit_file_dest
*.audit_file_dest='/u01/app/oracle/admin/db1/adump'

mkdir -p /u01/app/oracle/admin/db1/adump
```

10. If block change tracking was enabled on the original database, create the directory for the block change tracking file. This will be a directory under `db_create_file_dest`. Search the `spfile` for the name of the directory.

```
strings ${ORACLE_HOME}/dbs/spfile${ORACLE_SID}.ora | grep db_create_file_dest
*.db_create_file_dest='/u02/app/oracle/oradata/db1'

mkdir -p /u02/app/oracle/oradata/db1/<$ORA_UNQNAME if available or database name>/changetracking
```

11. Restart the instance with the restored server parameter file.

```
STARTUP FORCE NOMOUNT;
```

12. Restore the controlfile from the RMAN autobackup and mount the database.

```
set controlfile autobackup format for device type sbt to '%F';
run {
  allocate channel c1 device type sbt PARMS 'SBT_LIBRARY=/home/oracle/lib/libopc.so, SBT_PARMS=
(OPC_PFILE=/home/oracle/config)';
  restore controlfile from autobackup;
  alter database mount;
    }
```

13. Restore and recover the database.

```
RESTORE DATABASE;
RECOVER DATABASE;
```

14. RMAN will recover using archived redo logs until it can't find any more. It is normal for an error similar to the one below to occur when RMAN has applied the last archived redo log in the backup and can't find any more logs.

```
unable to find archived log
archived log thread=1 sequence=29
RMAN-00571: ===========================================================
RMAN-00569: =============== ERROR MESSAGE STACK FOLLOWS ===============
RMAN-00571: ===========================================================
RMAN-03002: failure of recover command at 06/28/2016 00:57:35
```

```
RMAN-06054: media recovery requesting unknown archived log for thread 1 with sequence 29 and
starting SCN of 2349563
```

15. Open the database with resetlogs.

```
ALTER DATABASE OPEN RESETLOGS;
```

The recovery is complete. The database will have all of the committed transactions as of the last backed up archived redo log.

Recovering a Database from a CLI Backup



Note

This topic is not applicable to Exadata DB Systems.

This topic explains how to perform a complete or point-in-time recovery of an existing database from a backup created with the dbcli create-backup command. The backup resides in the local Fast Recovery Area on the DB System.

To initiate the recovery, you'll use the Oracle Database CLI Reference command and specify the recovery type parameter (either `--recoverytype` or just `-t`). You can specify the following types of recovery:

- `-t Latest` for a complete recovery
- `-t SCN -s <scn>` for a recovery using a system change number (SCN) as the end point of the recovery
- `-t PITR <mm/dd/yyyy hh:mm:ss>` for a database point-in-time (incomplete) recovery based on a time stamp

The `dbcli create-recovery` attempts to perform a full recovery of the database. For information on performing a partial recovery (datafile, tablespace and PDB), see the *Oracle Database Backup Recovery Guide* for version 18.1, 12.2, 12.1, or 11.2.

PREREQUISITES

- The backup must have been created with the `dbcli create-backup` command.
- If the database is configured with Transparent Data Encryption (TDE), make sure the password-based and autologin TDE wallets are present in the following location:

```
/opt/oracle/dcs/commonstore/wallets/tde/<db_unique_name>
```

RECOVERING THE DATABASE

1. SSH to the DB System.

```
ssh -i <private_key_path> opc@<db_system_ip_address>
```

2. Log in as opc and then sudo to the root user. Use `sudo su -` with a hyphen to invoke the root user's profile, which will set the PATH to the dbcli directory (`/opt/oracle/dcs/bin`).

```
login as: opc

[opc@dbsys ~]$ sudo su -
```

3. Find the ID of database you want to recover by using the dbcli list-databases command. You'll need the ID for the following step.

```
[root@dbsys ~]# dbcli list-databases
ID                                       DB Name    DB Version CDB        Class    Shape
Storage    Status
---------------------------------------- ---------- ---------- ---------- -------- -------- -----
----- ----------
5a3e980b-e0fe-4909-9628-fcefe43b3326     prod       12.1.0.2   true       OLTP     odb1     ACFS
     Configured
```

4. Initiate the recovery by using the Oracle Database CLI Reference command and specifying the database ID, recovery type parameter (`-t`), and any parameter required for the recover type, like the time stamp or system change number.
   The following example initiates a complete recovery.

```
[root@dbsys ~]# dbcli create-recovery --dbid 5a3e980b-e0fe-4909-9628-fcefe43b3326 --recoverytype
Latest
{
```

```
  "jobId" : "c9f81228-2ce9-43b4-88f6-b260d398cf06",
  "status" : "Created",
  "message" : null,
  "reports" : [ ],
  "createTimestamp" : "August 08, 2016 18:20:47 PM UTC",
  "description" : "Create recovery for database id :5a3e980b-e0fe-4909-9628-fcefe43b3326",
  "updatedTime" : "August 08, 2016 18:20:47 PM UTC"
}
```

The following example initiates a point-in-time recovery of the specified database:

```
[root@dbsys ~]# dbcli create-recovery --dbid d4733796-dbea-4155-8606-24a85d64bd74 --recoverytype
PITR --recoveryTimeStamp 08/09/2016 5:12:15
```

Note the job ID in the command output.

5. Check the status of the recovery by using the dbcli describe-job command with the job ID from the previous step.

```
[root@dbsys ~]# dbcli describe-job -i c9f81228-2ce9-43b4-88f6-b260d398cf06

Job details
----------------------------------------------------------------
                     ID:  c9f81228-2ce9-43b4-88f6-b260d398cf06
            Description:  Create recovery for database id :5a3e980b-e0fe-4909-9628-fcefe43b3326
                 Status:  Success
                Created:  August 8, 2016 6:20:47 PM UTC
                Message:

Task Name                                Start Time                          End Time
               Status
---------------------------------------- ----------------------------------- --------------------
--------------- ----------
Database recovery validation             August 8, 2016 6:20:47 PM UTC       August 8, 2016
6:21:07 PM UTC       Success
Database recovery                        August 8, 2016 6:21:07 PM UTC       August 8, 2016
6:22:34 PM UTC       Success
enable block change tracking             August 8, 2016 6:22:34 PM UTC       August 8, 2016
6:22:35 PM UTC       Success
Open database                            August 8, 2016 6:22:35 PM UTC       August 8, 2016
6:22:44 PM UTC       Success
Restart database                         August 8, 2016 6:22:44 PM UTC       August 8, 2016
```

```
6:23:41 PM UTC       Success
Persist Recovery Metadata               August 8, 2016 6:23:41 PM UTC       August 8, 2016
6:23:41 PM UTC       Success
```

You can also check the database restore report logs on the DB System at:

```
/opt/oracle/dcs/log/<nodename>/rman/bkup/<db_unique_name>
```

Recovering a Database from the Oracle Cloud Infrastructure Classic Object Store



Note

This topic is not applicable to Exadata DB systems.

This topic explains how to recover a database using a backup created by the Oracle Database Backup Module and stored in Oracle Cloud Infrastructure Object Storage Classic.

The following terms are used throughout this topic:

- Source database: The database backup in Object Storage Classic.
- Target database: The new database on a DB system in Oracle Cloud Infrastructure.

PREREQUISITES

You'll need the following:

- The service name, identity name, container, user name, and password for Oracle Cloud Infrastructure Object Storage Classic.
- The backup password if password-based encryption was used when backing up to Object Storage Classic.
- The source database ID, database name, database unique name (required for setting up storage).
- If the source database is configured with Transparent Data Encryption (TDE), you'll need a backup of the wallet and the wallet password.
- Tnsnames to setup for any database links.

- The output of `Opatch lsinventory` for the source database Oracle_home, for reference.
- A copy of the sqlpatch directory from the source database home. This is required for rollback in case the target database does not include these patches.

Setting Up Storage on the DB System

1. SSH to the DB System.

```
ssh -i <private_key_path> opc@<db_system_ip_address>
```

2. Log in as opc and then sudo to the root user. Use `sudo su -` with a hyphen to invoke the root user's profile, which will set the PATH to the dbcli directory (`/opt/oracle/dcs/bin`).

```
login as: opc

[opc@dbsys ~]$ sudo su -
```

3. Use the dbcli create-dbstorage to set up directories for DATA, RECO, and REDO storage. The following example creates 10GB of ACFS storage for the tdetest database.

```
[root@dbsys ~]# dbcli create-dbstorage --dbname tdetest --dataSize 10 --dbstorage ACFS
```



Note

When migrating a version 11.2 database, ACFS storage must be specified.

4. Use the dbcli list-dbstorages command to list the storage ID. You'll need the ID for the next step.

```
[root@dbsys ~]# dbcli list-dbstorages
ID                                       Type   DBUnique Name        Status
---------------------------------------- ------ -------------------- ----------
9dcdfb8e-e589-4d5f-861a-e5ba981616ed     Acfs   tdetest              Configured
```

5. Use the dbcli describe-dbstorage command with the storage ID from the previous step to list the DATA, RECO and REDO locations.

```
[root@dbsys ~]# dbcli describe-dbstorage --id 9dcdfb8e-e589-4d5f-861a-e5ba981616ed
DBStorage details
----------------------------------------------------------------
                     ID: 9dcdfb8e-e589-4d5f-861a-e5ba981616ed
                DB Name: tdetest
          DBUnique Name: tdetest
         DB Resource ID:
           Storage Type: Acfs
          DATA Location: /u02/app/oracle/oradata/tdetest
          RECO Location: /u03/app/oracle/fast_recovery_area/
          REDO Location: /u03/app/oracle/redo/
                  State: ResourceState(status=Configured)
                Created: August 24, 2016 5:25:38 PM UTC
            UpdatedTime: August 24, 2016 5:25:53 PM UTC
```

6. Note the DATA, RECO and REDO locations. You'll need them later to set the `db_create_file_dest`, `db_create_online_log_dest`, and `db_recovery_file_dest` parameters for the database.

CHOOSING AN ORACLE_HOME

Decide which ORACLE_HOME to use for the database restore and then switch to that home with the correct ORACLE_BASE, ORACLE_HOME, and PATH settings. The ORACLE_HOME must not already be associated with a database.

To get a list of existing ORACLE_HOMEs and to ensure that the ORACLE_HOME is empty, use the dbcli list-dbhomes and the dbcli list-databases commands, respectively. To create a new ORACLE_HOME, use the dbcli create-dbhome command.

COPYING THE SOURCE DATABASE WALLETS

Skip this section if the source database is **not** configured with TDE.

1. On the DB system, become the oracle user:

```
sudo su - oracle
```

2. Create the following directory, if it does not already exist:

```
mkdir /opt/oracle/dcs/commonstore/wallets/tde/<db_unique_name>
```

3. Copy the ewallet.p12 file from the source database to the directory you created in the previous step.
4. On the target host, make sure that `$ORACLE_HOME/network/admin/sqlnet.ora` contains the following line:

```
ENCRYPTION_WALLET_LOCATION=(SOURCE=(METHOD=FILE)(METHOD_DATA=
(DIRECTORY=/opt/oracle/dcs/commonstore/wallets/tde/$ORACLE_UNQNAME)))
```

   Add the line if it doesn't exist in the file. (The line might not be there if this is a new home and no database has been created yet on this host.)
5. Create the autologin wallet from the password-based wallet to allow auto-open of the wallet during restore and recovery operations.

   For a version 12.1 or later database, use the `ADMINISTER KEY MANAGEMENT` command:

```
$cat create_autologin_12.sh

#!/bin/sh
if [ $# -lt 2 ]; then
        echo "Usage: $0 <dbuniquename><remotewalletlocation>"
        exit 1;
fi

mkdir /opt/oracle/dcs/commonstore/wallets/tde/$1
cp $2/ewallet.p12* /opt/oracle/dcs/commonstore/wallets/tde/$1
rm -f autokey.ora
echo "db_name=$1"  > autokey.ora
autokeystoreLog="autologinKeystore_`date +%Y%m%d_%H%M%S_%N`.log"
echo "Enter Keystore Password:"
read -s keystorePassword
echo "Creating AutoLoginKeystore -> "
sqlplus "/as sysdba"  <<EOF
spool $autokeystoreLog
set echo on
startup nomount pfile=autokey.ora
ADMINISTER KEY MANAGEMENT CREATE AUTO_LOGIN KEYSTORE
FROM KEYSTORE '/opt/oracle/dcs/commonstore/wallets/tde/$1' -- Keystore location
IDENTIFIED BY "$keystorePassword";
shutdown immediate;
EOF
```

Adjust the `cwallet.sso` permissions from `oracle:asmadmin` to `oracle:oinstall`.

```
$ ls -ltr /opt/oracle/dcs/commonstore/wallets/tde/<db_unique_name>

total 20

-rw-r--r-- 1 oracle oinstall 5680 Jul  6 11:39 ewallet.p12

-rw-r--r-- 1 oracle asmadmin 5725 Jul  6 11:39 cwallet.sso
```

For a version 11.2 database, use the `orapki` command:

```
orapki wallet create -wallet wallet_location -auto_login [-pwd password]
```

INSTALLING THE ORACLE DATABASE BACKUP MODULE

The backup module JAR file is included on the DB system but you need to install it.

1. SSH to the DB system, log in as opc, and then become the oracle user.

   ```
   ssh -i <path to SSH key used when launching the DB System> opc@<DB System IP address or hostname>

   sudo su - oracle
   ```

2. Change to the directory that contains the backup module `opc_install.jar` file.

   ```
   cd /opt/oracle/oak/pkgrepos/orapkgs/oss/<version>/
   ```

3. Use the command syntax described in Installing the Oracle Database Cloud Backup Module to install the backup module.

SETTING ENVIRONMENT VARIABLES

Set the following environment variables for the RMAN and SQL*Plus sessions for the database:

```
ORACLE_HOME=<path of Oracle Home where the database is to be restored>
ORACLE_SID=<database instance name>
ORACLE_UNQNAME=<db_unique_name in lower case>
NLS_DATE_FORMAT="mm/dd/yyyy hh24:mi:ss"
```

ALLOCATING AN RMAN SBT CHANNEL

For each restore operation, allocate an SBT channel and set the SBT_LIBRARY parameter to the location of the `libopc.so` file and the OPC_FILE parameter to the location of the `opc_`

sbt.ora file, for example:

```
ALLOCATE CHANNEL c1 DEVICE TYPE sbt MAXPIECESIZE 2 G FORMAT '%d_%I_%U' PARMS 'SBT_
LIBRARY=/tmp/oss/libopc.so ENV=(OPC_PFILE=/<ORACLE_HOME>/dbs/opc_sbt.ora)';
```

(For more information about these files, see Files Created When the Backup Module is Installed.)

Ensuring Decryption is Turned On

Make sure that decryption is turned on for all the RMAN restore sessions.

```
set decryption wallet open identified by <keystore password>;
```

For more information, see Providing Password Required to Decrypt Encrypted Backups.

Restoring Spfile

The following sample shell script restores the spfile. Set the $dbID variable to the dbid of the database being restored. By default, spfile is restored to $ORACLE_HOME/dbs/spfile<sid>.ora.

```
rman target / <<EOF

spool log to "`date +%Y%m%d_%H%M%S_%N`_dbid_${dbID}_restore_spfile.log"
startup nomount
set echo on
run {
ALLOCATE CHANNEL c1 DEVICE TYPE sbt MAXPIECESIZE 2 G FORMAT '%d_%I_%U' PARMS 'SBT_
LIBRARY=/tmp/oss/libopc.so ENV=(OPC_PFILE=/tmp/oss/opc_sbt.ora)';
SET DBID=$dbID;
RESTORE SPFILE FROM AUTOBACKUP;
shutdown immediate;
EOF
```

Setting the Database Parameters

1. Start the database in nomount mode.

   ```
   startup nomount
   ```

2. Update spfile and modify the following parameters.

- If the database storage type is ACFS, use the DATA, RECO, and REDO locations obtained from the `dbcli describe-dbstorage` command output, as described in Setting Up Storage on the DB System:

```
alter system set db_create_file_dest='/u02/app/oracle/oradata/' scope = spfile;
alter system set db_create_online_log_dest_1='/u03/app/oracle/redo' scope = spfile;
alter system set db_recovery_file_dest='/u03/app/oracle/fast_recovery_area' scope =
spfile;
```

- If the database storage type is ASM:

```
alter system set db_create_file_dest='+DATA' scope = spfile;
alter system set db_create_online_log_dest_1='+RECO' scope = spfile;
alter system set db_recovery_file_dest='+RECO' scope = spfile;
```

- Set `db_recovery_file_dest_size` is not set or is set incorrectly:

```
alter system set db_recovery_file_dest_size=<sizeG> scope=spfile;
```

- Set `audit_file_dest` to the correct value:

```
alter system set audit_file_dest=/u01/app/oracle/admin/<db_unique_name in lower
case>/adump
```

3. Remove the `control_files` parameter. The Oracle Managed Files (OMF) parameters will be used to create the control file.

```
alter system reset control_files scope=spfile;
```

4. Restart the database in nomount mode using the newly added parameters.

```
shutdown immediate
startup nomount
```

RESTORING THE CONTROL FILE

Modify the following sample shell script for your environment to restore the control file. Set the `$dbID` variable to the dbid of the database being restored. Set SBT_LIBRARY to the location specified in the `-libDir` parameter when you installed the Backup Module. Set OPC-PFILE to the location specified in the `-configFile` parameter, which defaults to *ORACLE_HOME*`/dbs/opc`*SID*`.ora`.

```
rman target / <<EOF

spool log to "`date +%Y%m%d_%H%M%S_%N`_dbid_${dbID}_restore_controlfile.log"
```

```
set echo on
run {
ALLOCATE CHANNEL c1 DEVICE TYPE sbt MAXPIECESIZE 2 G FORMAT '%d_%I_%U' PARMS 'SBT_LIBRARY=/<Backup
Module libDir>/libopc.so ENV=(OPC_PFILE=/<Backup Module configFile>/opcSID.ora)';
SET DBID=$dbID;
RESTORE CONTROLFILE FROM AUTOBACKUP;
alter database mount;
}

exit;
EOF
```

RESTORING THE DATABASE

1. Preview and validate the backup. The database is now mounted and RMAN should be able to locate the backup from the restored controlfile. This step helps ensure that the list of archivelogs is present and that the backup components can be restored .
   In the following examples, modify SBT_LIBRARY and OPC_PFILE as needed for your environment.

   ```
   rman target / <<EOF

   spool log to "`date +%Y%m%d_%H%M%S_%N`_restore_database_preview.log"
   set echo on
   run {
   ALLOCATE CHANNEL c1 DEVICE TYPE sbt MAXPIECESIZE 2 G FORMAT '%d_%I_%U' PARMS 'SBT_
   LIBRARY=/tmp/oss/libopc.so ENV=(OPC_PFILE=/tmp/oss/opc_sbt.ora)';
   ALLOCATE CHANNEL c2 DEVICE TYPE sbt MAXPIECESIZE 2 G FORMAT '%d_%I_%U' PARMS 'SBT_
   LIBRARY=/tmp/oss/libopc.so ENV=(OPC_PFILE=/tmp/oss/opc_sbt.ora)';
   ALLOCATE CHANNEL c3 DEVICE TYPE sbt MAXPIECESIZE 2 G FORMAT '%d_%I_%U' PARMS 'SBT_
   LIBRARY=/tmp/oss/libopc.so ENV=(OPC_PFILE=/tmp/oss/opc_sbt.ora)';
   restore database validate header preview;
   }
   ```

   Review the output and if there are error messages, investigate the cause of the problem.
2. Redirect the restore using `set newname` to restore the data files in OMF format and use `switch datafile all` to allow the control file to update with the new data file copies.

```
rman target / <<EOF

spool log to "`date +%Y%m%d_%H%M%S_%N`_restore_database_preview.log"
set echo on
run {
ALLOCATE CHANNEL c1 DEVICE TYPE sbt MAXPIECESIZE 2 G FORMAT '%d_%I_%U' PARMS 'SBT_
LIBRARY=/tmp/oss/libopc.so ENV=(OPC_PFILE=/tmp/oss/opc_sbt.ora)';
ALLOCATE CHANNEL c2 DEVICE TYPE sbt MAXPIECESIZE 2 G FORMAT '%d_%I_%U' PARMS 'SBT_
LIBRARY=/tmp/oss/libopc.so ENV=(OPC_PFILE=/tmp/oss/opc_sbt.ora)';
ALLOCATE CHANNEL c3 DEVICE TYPE sbt MAXPIECESIZE 2 G FORMAT '%d_%I_%U' PARMS 'SBT_
LIBRARY=/tmp/oss/libopc.so ENV=(OPC_PFILE=/tmp/oss/opc_sbt.ora)';
set newname for database to new;
restore database;
switch datafile all;
switch tempfile all;
recover database;
}
```

This recovery will attempt to use the last available archive log backup and then fail with an error, for example:

```
RMAN-00571: ===========================================================
RMAN-00569: =============== ERROR MESSAGE STACK FOLLOWS ===============
RMAN-00571: ===========================================================
RMAN-03002: failure of recover command at 07/20/2016 12:09:02
RMAN-06054: media recovery requesting unknown archived log for thread 1 with sequence 22 and
starting SCN of 878327
```

3. To complete the incomplete recovery, run a recovery using the sequence number and thread number shown in the RMAN-06054 message, for example:

```
Recover database until sequence 22 thread 1;
```

Resetting the Logs

Reset the logs.

```
alter database open resetlogs;
```

Preparing to Register the Database

Before you register the database:

1. Make sure the database COMPATIBLE parameter value is acceptable. If the value is less than the minimum, the database cannot be registered until you upgrade the database compatibility.
   The minimum compatibility values are as follows:
   - For a version 18.1 database - 18.0.0.0
   - For a version 12.2 or 12.1 database - 12.1.0.2
   - For a version 11.2 database - 11.2.0.4
2. Verify that the database has registered with the listener and the service name.

   ```
   lsnrctl services
   ```

3. Make sure the password file was restored or created for the new database.

   ```
   ls -ltr $ORACLE_HOME/dbs/orapw<oracle sid>
   ```

   If the file does not exist, create it using the orapwd utility.

   ```
   orapwd file=<$ORACLE_HOME/dbs/orapw<$ORACLE_SID>> password=<sys password>
   ```

4. Make sure the restored database if open in read write mode.

   ```
   select open_mode from v$database;
   ```

   The command output should indicate read write mode. The `dbcli register-database` command will attempt to run datapatch, which requires read write mode. If there are PDBs, they should also be in read write mode to ensure that datapatch runs on them.
5. From oracle home on the restored database, use the following command verify the connection to SYS:

   ```
   conn sys/<password>@//<hostname>:1521/<database service name>
   ```

   This connection is required to register the database later. Fix any connection issues before continuing.
6. Make sure the database is running on spfile by using the SQL*Plus command.

   ```
   SHOW PARAMETERS SPFILE
   ```

7. (Optional) If you would like to manage the database backup with the dbcli command line interface, you can associate a new or existing backup configuration with the migrated

database when you register it or after you register it. A backup configuration defines the backup destination and recovery window for the database. Use the following commands to create, list, and display backup configurations:

- dbcli update-backupconfig
- dbcli list-backupconfigs
- dbcli describe-backupconfig

8. Copy the folder `$ORACLE_HOME/sqlpatch` from source database to the target database. This will enable the `dbcli register-database` command to roll back any conflicting patches.



Note

If you are migrating a version 11.2 database, additional steps are required after you register the database. For more information, see Rolling Back Patches on a Version 11.2 Database.

Registering the Database on the DB System

The dbcli register-database command registers the restored database to the dcs-agent so it can be managed by the dcs-agent stack.



Note

The `dbcli register-database` command is not available on 2-node RAC DB systems.

As the root user, use the `dbcli register-database` command to register the database on the DB system, for example:

```
[root@dbsys ~]# dbcli register-database --dbclass OLTP --dbshape odb1 --servicename tdetest --
syspassword
Password for SYS:
```

```
{
  "jobId" : "317b430f-ad5f-42ae-bb07-13f053d266e2",
  "status" : "Created",
  "message" : null,
  "reports" : [ ],
  "createTimestamp" : "August 08, 2016 05:55:49 AM EDT",
  "description" : "Database service registration with db service name: tdetest",
  "updatedTime" : "August 08, 2016 05:55:49 AM EDT"
}
```

UPDATING TNSNAMES.ORA

Check the `tnsnames.ora` in the backup location, check the database links used in the cloned database, and then add any relevant connection strings to the cloned database file at `$ORACLE_HOME/network/admin/tnsnames.ora`.

ROLLING BACK PATCHES ON A VERSION 11.2 DATABASE

For version 11.2 databases, the sqlpatch application is not automated, so any interim patches (previously known as a "one-off" patches) applied to the source database that are not part of the installed PSU must be rolled back manually in the target database. After registering the database, execute the `catbundle.sql` script and then the `postinstall.sql` script with the corresponding PSU patch (or the overlay patch on top of the PSU patch), as described below.



Tip

Some interim patches may include files written to the $ORACLE_HOME/rdbms/admin directory as well as the $ORACLE_HOME/sqlpatch directory. Oracle recommends that you roll back these patches in the source database using the instructions in the patch read-me prior to migrating the database to OCI environment. Contact Oracle Support if you need assistance with rolling back these patches.

1. On the DB System, use the `dbcli list-dbhomes` command to find the PSU patch number for the version 11.2 database home. In the following sample command output, the PSU patch number is the second number in the DB Version column:

```
[root@dbsys ~]# dbcli  list-dbhomes
ID                                   Name               DB Version
Home Location                             Status
------------------------------------ -----------------  ------------------------------------  --
----------------------------------------- ----------
59d9bc6f-3880-4d4f-b5a6-c140f16f8c64 OraDB11204_home1  11.2.0.4.160719 (23054319, 23054359)
/u01/app/oracle/product/11.2.0.4/dbhome_1 Configured
```

   (The first patch number, 23054319 in the example above, is for the OCW component in the database home.)

2. Find the overlay patch, if any, by using the `lsinventory` command. In the following example, patch number *24460960* is the overlay patch on top of the 23054359 PSU patch.

```
$ $ORACLE_HOME/OPatch/opatch lsinventory
...
Installed Top-level Products (1):

Oracle Database 11g                                                  11.2.0.4.0
There are 1 products installed in this Oracle Home.


Interim patches (5) :

Patch  24460960     : applied on Fri Sep 02 15:28:17 UTC 2016
Unique Patch ID:  20539912
   Created on 31 Aug 2016, 02:46:31 hrs PST8PDT
   Bugs fixed:
     23513711, 23065323, 21281607, 24006821, 23315889, 22551446, 21174504
   This patch overlays patches:
     23054359
   This patch needs patches:
     23054359
   as prerequisites
```

3. Start SQL*Plus and execute the `catbundle.sql` script, for example:

```
SQL> startup
SQL> connect / as sysdba
SQL> @$ORACLE_HOME/rdbms/admin/catbundle.sql psu apply
exit
```

4. Apply the sqlpatch, using the overlay patch number from the previous step, for example:

```
SQL> connect / as sysdba
SQL> @$ORACLE_HOME/sqlpatch/24460960/postinstall.sql
exit
```



Note

If the source database has one-off patches installed and those patches are **not** part of the installed PSU in the cloud environment, then the SQL changes that correspond to those one-off patches need to be rolled back. To rollback the SQL changes, copy the `$ORACLE_HOME/sqlpatch/<patch#>/postdeinstall.sql` script from the source environment to the cloud environment and execute the `postdeinstall.sql` script.

POST RESTORE CHECKLIST

After the database is restored and registered on the DB system, use the following checklist to verify the results and perform any post-restore customizations.

1. Make sure the database files were restored in OMF format.
2. Make sure the database is listed in the dbcli list-databases command output.
3. Check for the following external references in the database and update them if necessary:

- External tables: If the source database uses external tables, back up that data and migrate it to the target host.
- Directories: Customize the default directories as needed for the restored database.
- Database links: Make sure all the required TNS entries are updated in the `tnsnames.ora` file in ORACLE_HOME.
- Email and URLs: Make sure any email addresses and URLs used in the database are still accessible from the DB system.
- Scheduled jobs: Review the jobs scheduled in source database and schedule similar jobs as needed in the restored database.

4. If you associated a backup configuration when you registered the database, run a test back up using the dbcli create-backup command.
5. If the restored database contains a CDB and PDBs, verify that patches have been applied to all PDBs.

Using Oracle Data Guard



Note

This topic is not applicable to Exadata DB systems.

This topic explains how to use the Console to manage Data Guard associations in your DB system. To configure a Data Guard system across regions or between on-premises and Oracle Cloud Infrastructure DB systems, you must access the database host directly and use the DGMGRL utility.

For complete information on Oracle Data Guard, see the Data Guard Concepts and Administration documentation on the Oracle Document Portal.

Required IAM Service Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

If you're new to policies, see Getting Started with Policies and Common Policies.

Prerequisites

A Data Guard implementation requires two DB systems, one containing the primary database and one containing the standby database. When you enable Data Guard for a virtual machine DB system database, a new DB system with the standby database is created and associated with the primary database. For a bare metal DB system, the DB system with the database to be used as the standby must already exist before you enable Data Guard.



Tip

A Data Guard configuration on the Oracle Cloud Infrastructure is limited to one standby database per primary database.

Requirement details are as follows:

- Both DB systems must be in the same compartment, and they must be the same shape.
- The database versions and editions must be identical. Data Guard does not support Standard Edition. (Active Data Guard requires Enterprise Edition - Extreme Performance.)
- The database version determines whether Active Data Guard is enabled. If you are using the BYOL licensing model and if your license does not include Active Data Guard, you must either use Enterprise Edition - High Performance or set up Data Guard manually. See Using Oracle Data Guard with the Database CLI.

- Both DB systems must use the same VCN, and port 1521 must be open.
- **Important!** Properly configure the security list ingress and egress rules for the subnets of both DB systems in the Data Guard association to allow TCP traffic to flow between the applicable ports. Ensure that the rules you create are stateful (the default).

  For example, if the subnet of the primary DB System uses the source CIDR 10.0.0.0/24 and the subnet of the standby DB system uses the source CIDR 10.0.1.0/24, create rules as shown in the following example.

  Note

  The egress rules in the example show how to enable TCP traffic only for port 1521, which is a minimum requirement for Data Guard to work. If TCP traffic is already enabled on all of your outgoing ports (0.0.0.0/0), then you need not explicitly add these specific egress rules.

  Security List for Primary DB System's Subnet

```
Ingress Rules:

Stateless: No
Source: 10.0.1.0/24
IP Protocol: TCP
Source Port Range: All
Destination Port Range: 1521
Allows: TCP traffic for ports: 1521

Egress Rules:

Stateless: No
Destination: 10.0.1.0/24
IP Protocol: TCP
Source Port Range: All
Destination Port Range: 1521
Allows: TCP traffic for ports: 1521
```

Security List for Standby DB System's Subnet

```
Ingress Rules:

Stateless: No
Source: 10.0.0.0/24
IP Protocol: TCP
Source Port Range: All
Destination Port Range: 1521
Allows: TCP traffic for ports: 1521

Egress Rules:

Stateless: No
Destination: 10.0.0.0/24
IP Protocol: TCP
Source Port Range: All
Destination Port Range: 1521
Allows: TCP traffic for ports: 1521
```

For information about creating and editing rules, see Security Lists.

Availability Domain and Fault Domain Considerations for Data Guard

Oracle recommends that the DB system of the standby database be in a different availability domain from the DB system of the primary database to improve availability and disaster recovery. If you enable Data Guard for a database and your standby database is in the same availability domain as the primary (either by choice, or because you are working in a single availability domain region), Oracle recommends that you place the standby database in a different fault domain from that of the primary database. Note that if your primary and standby databases are 2-node RAC databases and both are in the same availability domain, only one of the two nodes of the standby database can be in a fault domain that does not include any other nodes from either the primary or standby database. This is because each availability domain has only three fault domains, and the primary and standby databases have a combined total of 4 nodes. For more information on availability domains and fault domains, see Regions and Availability Domains.

Working with Data Guard

Oracle Data Guard ensures high availability, data protection, and disaster recovery for enterprise data. The Oracle Cloud Infrastructure Database Data Guard implementation requires two databases, one in a primary role and one in a standby role. The two databases compose a Data Guard association. Most of your applications access the primary database. The standby database is a transactionally consistent copy of the primary database.

Data Guard maintains the standby database by transmitting and applying redo data from the primary database. If the primary database becomes unavailable, you can use Data Guard to switch or fail over the standby database to the primary role.



Tip

The standby databases in Oracle Cloud Infrastructure Database are physical standbys.

SWITCHOVER

A switchover reverses the primary and standby database roles. Each database continues to participate in the Data Guard association in its new role. A switchover ensures no data loss. You can use a switchover before you perform planned maintenance on the primary database.

FAILOVER

A failover transitions the standby database into the primary role after the existing primary database fails or becomes unreachable. A failover might result in some data loss when you use **Maximum Performance** protection mode.

REINSTATE

Reinstates a database into the standby role in a Data Guard association. You can use the reinstate command to return a failed database into service after correcting the cause of failure.



Note

You can't terminate a primary database that has a Data Guard association with a peer (standby) database. Delete the standby database first. Alternatively, you can perform a switchover to the standby database, and then terminate the primary database.

You can't terminate a DB system that includes Data Guard enabled databases. To remove the Data Guard association:

- For a bare metal DB system database - terminate the standby database.
- For a virtual machine DB system database - terminate the standby DB system.

Using the Console

The Console allows you to enable a Data Guard association between databases, change the role of a database in a Data Guard association using either a *switchover* or a *failover* operation, and *reinstate* a failed database.

When you enable Data Guard, a separate Data Guard association is created for the primary and the standby database.

To enable Data Guard on a bare metal DB system

If you don't already have bare metal DB systems with the databases that will assume the primary and standby roles, create them as described in Managing Bare Metal and Virtual Machine DB Systems. A new DB system includes an initial database.

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose the **Compartment** that contains the DB system with the database for which you want to enable Data Guard.
3. Click the name of the DB system that contains the database you want to assume the primary role, and then click the name of that database.



Tip

If Data Guard is already enabled, a shield icon appears next to the database name.

4. Under **Resources**, click **Data Guard Associations**.
5. Click **Enable Data Guard**.
6. In the **Enable Data Guard** dialog box, configure your Data Guard association.
   - **Peer Database Availability Domain:** (Informational) Shows the availability domain of the selected peer DB system. If your database is in a single availability domain region, or if you choose to provision your peer (standby) database in the same availability domain as you primary database, the system provides a fault domain selector for your peer database. Oracle recommends that your peer DB system be in a different fault domain from your primary DB system in such cases. For more information on fault domains, see Fault Domains.
   - **Peer DB System:** Select the DB system that will contain the peer (standby) database.
   - **Protection Mode:** The protection mode used for this Data Guard association. The Console supports only **Maximum Performance**.
   - **Transport Type:** The redo transport type used for this Data Guard association. The Console supports only **Async**.
   - **Database Admin Password:** Enter the primary database admin password. The same password is used for the standby database.

Confirm Database Admin Password: Re-enter the Database Admin Password you specified.

7. Click **Enable**.
   When the association is created, a shield icon appears next to the name of this database and its peer, and their respective roles (primary or standby) are displayed.

To enable Data Guard on a virtual machine DB system

If you don't already have a virtual machine DB system with the database that will assume the primary role, create it as described in Managing Bare Metal and Virtual Machine DB Systems. The new DB system will include the initial database. When you enable Data Guard on the primary database, a new virtual machine DB system will be launched for the standby database.

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose the **Compartment** that contains the DB system with the database for which you want to enable Data Guard.
3. Click the name of the DB system that contains the database you want to assume the primary role, and then click the name of that database.



Tip

If Data Guard is already enabled, a shield icon appears next to the database name.

4. Under **Resources**, click **Data Guard Associations**.
5. Click **Enable Data Guard**.
6. In the **Enable Data Guard** dialog box, configure your Data Guard association.
   Display Name: A friendly, display name for the DB system. The name doesn't need to be unique. An Oracle Cloud Identifier (OCID) will uniquely identify the DB system.

Availability Domain: The availability domain in which the DB system resides.

- **Virtual Cloud Network:** (Informational) Shows the VCN in which the DB system will be launched. The VCN of the primary database and the standby database must be the same.
- **Client Subnet:** The subnet to which the DB system should attach.
  Do not use a subnet that overlaps with 192.168.16.16/28, which is used by the Oracle Clusterware private interconnect on the database instance. Specifying an overlapping subnet will cause the private interconnect to malfunction.
- **Hostname Prefix:** Your choice of host name for the DB system. The host name must begin with an alphabetic character, and can contain only alphanumeric characters and hyphens (-). The maximum number of characters allowed for a virtual machine DB system is 16.

  Important

  The host name must be unique within the subnet. If it is not unique, the DB system will fail to provision.

- **Host Domain Name:** The domain name for the DB system. If the selected subnet uses the Oracle-provided Internet and VCN Resolver for DNS name resolution, this field displays the domain name for the subnet and it can't be changed. Otherwise, you can provide your choice of a domain name. Hyphens (-) are not permitted.

  Host and Domain URL: Combines the host and domain names to display the fully qualified domain name (FQDN) for the database. The maximum length is 64 characters.
- **Protection Mode:** The protection mode used for this Data Guard association. The Console supports only **Maximum Performance**.

- **Transport Type:** The redo transport type used for this Data Guard association. The Console supports only **Async**.
- **Database Admin Password:** Enter the primary database admin password. The same password is used for the standby database.

  Confirm Database Admin Password: Re-enter the Database Admin Password you specified.

7. Click **Enable**.
   When the association is created, a shield icon appears next to the name of this database and its peer, and their respective roles (primary or standby) are displayed.

To perform a database switchover

You initiate a switchover operation by using the Data Guard association of the primary database.

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose the **Compartment** that contains the DB system with the primary database you want to switch over.
3. Click the DB system name, and then click the name of the primary database.
4. Under **Resources**, click **Data Guard Associations**.
5. For the Data Guard association on which you want to perform a switchover, click the Actions icon (three dots), and then click **Switchover**.
6. In the **Switchover Database** dialog box, enter the database admin password, and then click **OK**.
   This database should now assume the role of the standby, and the standby should assume the role of the primary in the Data Guard association.

To perform a database failover

You initiate a failover operation by using the Data Guard association of the standby database.

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose the **Compartment** that contains the DB system with the primary database's peer standby you want to fail over to.
3. Click the DB system name, and then click the name of the standby database.
4. Under **Resources**, click **Data Guard Associations**.
5. For the Data Guard association on which you want to perform a failover, click **Failover**.
6. In the **Failover Database** dialog box, enter the database admin password, and then click **OK**.
   This database should now assume the role of the primary, and the old primary's role should display as **Disabled Standby**.

To reinstate a database

After you fail over a primary database to its standby, the standby assumes the primary role and the old primary is identified as a disabled standby. After you correct the cause of failure, you can reinstate the failed database as a functioning standby for the current primary by using its Data Guard association.



Note

Before you can reinstate a version 12.2 database, you must perform some steps on the database host to stop the database or start it in `MOUNT` mode.

Set your ORACLE_UNQNAME environment variable to the value of the Database Unique Name (as seen in the Console), and then run these commands:

```
srvctl stop database -d db-unique-name -o abort
srvctl start database -d db-unique-name -o mount
```

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose the **Compartment** that contains the DB system with the failed database you want to reinstate.
3. Click the DB system name, and then click the database name.
4. Under **Resources**, click **Data Guard Associations**.
5. For the Data Guard association on which you want to reinstate this database, click the Actions icon (three dots), and then click **Reinstate**.
6. In the **Reinstate Database** dialog box, enter the database admin password, and then click **OK**.
   This database should now be reinstated as the standby in the Data Guard association.

To terminate a Data Guard association on a bare metal DB system

On a bare metal DB system, you remove a Data Guard association by terminating the standby database.

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose the **Compartment** that contains the DB system that includes the standby database you want to terminate.
3. Click the DB system name.
4. For the standby database you want to terminate, click the Actions icon (three dots), and then click **Terminate**.
5. In the **Terminate Database** dialog box, enter the name of the database, and then click **OK**.

To terminate a Data Guard association on a virtual machine DB system

On a virtual machine DB system, you remove a Data Guard association by terminating the standby DB system.

1. Open the navigation menu. Under **Database**, click **Bare Metal, VM, and Exadata**.
2. Choose the **Compartment** that contains the standby DB system that you want to terminate.
3. Click the DB system name, click the Actions icon (three dots), and then click **Terminate**.
4. Confirm when prompted.
   The DB system's icon indicates Terminating.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage Data Guard associations:

- CreateDataGuardAssociation
- GetDataGuardAssociation
- ListDataGuardAssociations
- SwitchoverDataGuardAssociation
- FailoverDataGuardAssociation
- ReinstateDataGuardAssociation
- DeleteDbHome - To terminate a bare metal DB system Data Guard association, delete the standby database.
- TerminateDbSystem - To terminate a virtual machine DB system Data Guard association, terminate the standby DB system.

For the complete list of APIs for the Database service, see Database Service API.

Using Oracle Data Guard with the Database CLI

Oracle recommends that you use the Console instead of the database CLI to set up and work with Data Guard in Oracle Cloud Infrastructure. See Using Oracle Data Guard for information and instructions.



Note

This topic is not applicable to Exadata DB systems. You can manually configure Data Guard on Exadata DB systems using native Oracle Database utilities and commands, however this topic explains how set up primary and standby databases using `dbcli`, which is not available on Exadata DB systems. For more information, see *Data Guard Concepts and Administration* for version 18.1, 12.2, 12.1, or 11.2.

This topic explains how to use the database CLI to set up Data Guard with Fast-Start Failover (FSFO) in Oracle Cloud Infrastructure. The following sections explain how to prepare the primary and standby databases, and then configure Data Guard to transmit redo data from the primary database and apply it to the standby database.



Note

This topic assumes that you are familiar with Data Guard and FSFO. To learn more about them, see documentation at the Oracle Document Portal.

Prerequisites

To perform the procedures in this topic, you'll need the following information for the primary and standby databases.

- db_name (or oracle_sid)
- db_unique_name
- oracle home directory (or database home)

To find the database information

After you've launched the primary and standby DB systems and created databases as described later in this topic, you can use the CLI on those systems to find the needed database information.

1. SSH to the DB System.

```
ssh -i <private_key_path> opc@<db_system_ip_address>
```

2. Log in as opc and then sudo to the root user. Use `sudo su -` with a hyphen to invoke the root user's profile, which will set the PATH to the dbcli directory (`/opt/oracle/dcs/bin`).

```
login as: opc

[opc@dbsys ~]$ sudo su -
```

3. To find the db_name (or oracle_sid) and db_uniqueName, run the `dbcli list-databases -j` command.

```
[root@dbsys ~]# dbcli list-databases -j
[ {
  "id" : "80ad855a-5145-4f8f-a08f-406c5e4684ff",
  "name" : "dbtst",
  "dbName" : "dbtst",
  "databaseUniqueName" : "dbtst_phx1cs",
  "dbVersion" : "12.1.0.2",
  "dbHomeId" : "2efe7af7-0b70-4e9b-ba8b-71f11c6fe287",
```

```
  "instanceOnly" : false,
    .
    .
    .
```

4. To find the oracle home directory (or database home), run the `dbcli list-dbhomes` command. If there are multiple database homes on the DB system, use the one that matches the "dbHomeId" in the `dbcli list-databases -j` command output shown above.

```
[root@dbtst ~]# dbcli list-dbhomes

ID                                       Name                 DB Version
     Home Location                                 Status
---------------------------------------- -------------------- --------------------------------
----- ------------------------------------------- ----------
2efe7af7-0b70-4e9b-ba8b-71f11c6fe287     OraDB12102_home1     12.1.0.2.160719 (23739960,
23144544)     /u01/app/oracle/product/12.1.0.2/dbhome_1     Configured
33ae99fe-5413-4392-88da-997f3cd24c0f     OraDB11204_home1     11.2.0.4.160719 (23054319,
23054359)     /u01/app/oracle/product/11.2.0.4/dbhome_1     Configured
```

Creating a Primary DB System

If you don't already have a primary DB system, create one as described in Managing Bare Metal and Virtual Machine DB Systems. The DB system will include an initial database. You can create additional databases by using the dbcli create-database command available on the DB system.

Creating a Standby DB System



Note

The standby database must have the same db_name as the primary database, but it must have a different db_unique_name. If you use the same database name for the standby and primary, you will have to delete the database from the standby DB system by using the `dbcli delete-database` command before you can run the `dbcli create-database` command described below. Deleting and creating the database will take several minutes to complete. The `dbcli` commands must be run as the root user.

1. Create a standby DB system as described in Managing Bare Metal and Virtual Machine DB Systems and wait for the DB system to finish provisioning and become available. You can create the standby DB system in a different availability domain from the primary DB system for availability and disaster recovery purposes (this is strongly recommended). You can create the standby DB system in the primary DB system's cloud network so that both systems are in a single, routable network.
2. SSH to the DB System.

   ```
   ssh -i <private_key_path> opc@<db_system_ip_address>
   ```

3. Log in as opc and then sudo to the root user. Use `sudo su -` with a hyphen to invoke the root user's profile, which will set the PATH to the dbcli directory (`/opt/oracle/dcs/bin`).

   ```
   login as: opc

   [opc@dbsys ~]$ sudo su -
   ```

4. The DB system will include an initial database, but you'll need to create a standby database by using the `dbcli create-database` command with the `--instanceonly`

parameter. This parameter creates only the database storage structure and starts the database in nomount mode (no other database files are created).

When using `--instanceonly`, both the `--dbname` and `--adminpassword` parameters are required and they should match the dbname and admin password of the primary database to avoid confusion.

The following sample command prompts for the admin password and then creates a storage structure for a database named dbname.

```
[root@dbsys ~]# dbcli create-database --dbname <same as primary dbname> --databaseUniqueName
<different from primary uniquename> --instanceonly --adminpassword
```

If you are using pluggable databases, also specify the `--cdb` parameter.

For complete command syntax, see dbcli create-database.

5. Wait a few minutes for the `dbcli create-database` command to create the standby database.

   You can use the `dbcli list-jobs` command to verify that the creation job ran successfully, and then the `dbcli list-databases` command verify that the database is configured.

Preparing the Primary DB System

To prepare the primary DB system, you'll need to configure static listeners, update tnsnames.ora, and configure some database settings and parameters.

CONFIGURING THE STATIC LISTENERS

Create static listeners to be used by RMAN and Data Guard Broker.

1. SSH to the primary DB system, log in as the opc or root user, and sudo to the grid OS user.

   ```
   sudo su - grid
   ```

2. Edit `/u01/app/<version>/grid/network/admin/listener.ora` and add the following content to it. The first static listener shown here is optional. The second `DGMGRL` static listener is optional for version 12.1 or later databases, but required for version 11.2 databases.

```
SID_LIST_LISTENER=
  (SID_LIST=
    (SID_DESC=
    (SDU=65535)
    (GLOBAL_DBNAME = <primary_db_unique_name>.<primary_db_domain>)
    (SID_NAME = <primary_oracle_sid>)
    (ORACLE_HOME=<oracle_home_directory>)
    (ENVS="TNS_ADMIN=<oracle_home_directory>/network/admin")
    )
    (SID_DESC=
    (SDU=65535)
       (GLOBAL_DBNAME = <primary_db_unique_name>_DGMGRL.<primary_db_domain>)
    (SID_NAME = <primary_oracle_sid>)
    (ORACLE_HOME=<oracle_home_directory>)
    (ENVS="TNS_ADMIN=<oracle_home_directory>/network/admin")
    )
  )
```

3. Save your changes and then restart the listener.

```
$ srvctl stop listener
$ srvctl start listener
```

ADDING NET SERVICE NAMES TO TNSNAMES.ORA

As the oracle user, edit `$ORACLE_HOME/network/admin/tnsnames.ora` and add the standby database net service name to it.

```
<standby db_unique_name> =
  (DESCRIPTION =
    (SDU=65535)
    (ADDRESS = (PROTOCOL = TCP)(HOST = <standby_server>.<domain>) (PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = <standby db_unique_name>.<standby db_domain>)
    )
  )
```

The sample above assumes that name resolution is working and that the *<standby_server>.<domain>* is resolvable at the primary database. You can also use the private IP address of the standby server if the IP addresses are routable within a single cloud network (VCN).

CONFIGURING PRIMARY DATABASE PARAMETERS



Tip

If the primary and standby hosts have different directory structures, you might need to set additional parameters that are not discussed here, such as the `log_file_name_convert` parameter. See the RMAN documentation for more information about how to create standbys for hosts with different directory structures.

1. As the oracle user, enable automatic standby file management.

```
SQL> alter system set standby_file_management=AUTO;
```

2. Identify the Broker configuration file names and locations. The commands used for this depend on the type of database storage. If you're not sure of the database storage type, use the dbcli list-databases command on the DB system.

   For ACFS database storage, use the following commands to set the Broker configuration files.

```
SQL> alter system set dg_broker_config_file1='/u02/app/oracle/oradata/<Primary db_unique_
name>/dbs/dr1<Primary db_unique_name>.dat';
SQL> alter system set dg_broker_config_file2='/u02/app/oracle/oradata/<Primary db_unique_
name>/dbs/dr2<Primary db_unique_name>.dat';
```

   For ASM database storage, use the following commands to set the Broker configuration files.

```
SQL> alter system set dg_broker_config_file1='+DATA/<Primary db_unique_name>/dr1<db_unique_
name>.dat';
SQL> alter system set dg_broker_config_file2='+DATA/<Primary db_unique_name>/dr2<db_unique_
name>.dat';
```

3. Enable Broker DMON process for the database.

```
SQL> alter system set dg_broker_start=true;
```

4. Force database logging for all database transactions.

```
SQL> alter database force logging ;
```

5. Add Standby Redo Logs (SRLs), based on the Online Redo Logs (ORLs). On a newly launched DB system, there will be three ORLs of size 1073741824, so create four SRLs of the same size.

   You can use the query below to determine the number and size (in bytes) of the ORLs.

```
SQL> select group#, bytes from v$log;

    GROUP# BYTES
---------- ----------
         1 1073741824
         2 1073741824
         3 1073741824
```

   All of the ORLs must be the same size.

   The SRLs must be the same size as the ORLs, but there must be at least one more SRL than the ORLs. In the example above, there are three ORLs, so four SRLs are required. So specify the current redo logs plus one, and use the same size as the redo logs.

```
SQL> alter database add standby logfile thread 1 size <size>;
```

   There should be only one member in the SRL group (by default, a DB system is created with only one member per SRL group). To ensure this, you can name the file with the following syntax.

```
alter database add standby logfile thread 1 group 4 (<logfile name with full path>) size
1073741824, group 5(<logfile name with full path>) size 1073741824 ...
```

   For ASM/OMF configurations, the above command uses the diskgroup instead of *<logfile name with full path>*.

```
alter database add standby logfile thread 1 group 4 (+RECO) size 1073741824, group 5(+RECO) size
1073741824 ...
```



Tip

ORLs and SRLs should be sized so that log switches do not occur more frequently than every 10 minutes. This requires knowledge of the application and may need to be adjusted after deployment. For more information, see Use Standby Redo Logs and Configure Size Appropriately.

6. Verify that you created the correct number of SRLs.

```
SQL> select group#, bytes from v$standby_log;
```

7. Make sure the database is in ARCHIVELOG mode.

```
SQL> archive log list
```

8. Enable database FLASHBACK. The minimum recommended value for db_flashback_retention_target is 120 minutes.

```
SQL> alter database flashback on ;
SQL> alter system set db_flashback_retention_target=120;
```

9. Perform a single switch redo log to activate archiving if database is newly created. (At least one log must be archived prior to running the RMAN duplicate.)

```
SQL> alter system switch logfile;
```

Preparing the Standby Database

Before you prepare the standby database, make sure the database home on the standby is the same version as on the primary. (If the primary and standby databases are both newly created with the same database version, the database homes will be the same.) If it is not, create a database home that is the same version. You can use the dbcli list-dbhomes command to verify the versions and the dbcli create-dbhome command to create a new database home as needed.

To prepare the standby DB system, you'll need to configure static listeners, update tnsnames.ora, configure TDE Wallet, create a temporary password file, verify connectivity, run RMAN DUPLICATE, enable FLASHBACK, and then create the database service.

CONFIGURING THE STATIC LISTENERS

Create static listeners to be used by RMAN and Data Guard Broker.

1. SSH to the standby DB system, log in as the opc or root user, and sudo to the grid OS user.

   ```
   sudo su - grid
   ```

2. Append the following content to `/u01/app/<db_version>/grid/network/admin/listener.ora`.

   The first static listener shown below is required for RMAN DUPLICATE. The second DGMGRL static listener is optional for database versions 12.2.0.1 and 12.1.0.2, but required for database version 11.2.0.4.

   ```
   SID_LIST_LISTENER=
     (SID_LIST=
       (SID_DESC=
       (SDU=65535)
       (GLOBAL_DBNAME = <standby db_unique_name>.<standby db_domain>)
       (SID_NAME = <standby oracle_sid>)
       (ORACLE_HOME=<oracle home directory>)
       (ENVS="TNS_ADMIN=<oracle home directory>/network/admin")
       )
       (SID_DESC=
       (SDU=65535)
       (GLOBAL_DBNAME = <standby db_unique_name>_DGMGRL.<standby db_domain>)
       (SID_NAME = <standby oracle_sid>)
       (ORACLE_HOME=<oracle home directory>)
       (ENVS="TNS_ADMIN=<oracle home directory>/network/admin")
       )
     )
   ```

3. Restart the listener.

   ```
   $ srvctl stop listener
   $ srvctl start listener
   ```

4. Verify that the static listeners are available. The sample output below is for database version 12.1.0.2. Note that the `...status UNKNOWN` messages are expected at this point.

```
$ lsnrctl status

LSNRCTL for Linux: Version 12.1.0.2.0 - Production on 29-SEP-2016 21:09:25

Copyright (c) 1991, 2014, Oracle.  All rights reserved.

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=LISTENER)))
STATUS of the LISTENER
------------------------
Alias                     LISTENER
Version                   TNSLSNR for Linux: Version 12.1.0.2.0 - Production
Start Date                29-SEP-2016 21:09:19
Uptime                    0 days 0 hr. 0 min. 5 sec
Trace Level               off
Security                  ON: Local OS Authentication
SNMP                      OFF
Listener Parameter File   /u01/app/12.1.0.2/grid/network/admin/listener.ora
Listener Log File         /u01/app/grid/diag/tnslsnr/dg2/listener/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=LISTENER)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=10.0.1.24)(PORT=1521)))
Services Summary...
Service "dg2_phx2hx.oratst.org" has 1 instance(s).
  Instance "dg2", status UNKNOWN, has 1 handler(s) for this service...
Service "dg2_phx2hx_DGMGRL.oratst.org" has 1 instance(s).
  Instance "dg2", status UNKNOWN, has 1 handler(s) for this service...
The command completed successfully
```

ADDING NET SERVICE NAMES TO TNSNAMES.ORA

As the oracle user, add the standby database net service name to `$ORACLE_HOME/network/admin/tnsnames.ora`. $ORACLE_HOME is the database home where the standby database is running.

```
<Primary db_unique_name> =
  (DESCRIPTION =
    (SDU=65535)
    (ADDRESS = (PROTOCOL = TCP)(HOST = <primary_server>.<domain>) (PORT = 1521))
```

```
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = <primary db_unique_name).<primary db_domain>)
    )
  )

<Standby db_unique_name> =
  (DESCRIPTION =
    (SDU=65535)
    (ADDRESS = (PROTOCOL = TCP)(HOST = <standby_server>.<domain>) (PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = <standby db_unique_name>.<db_domain>)
    )
  )
```

Copying the TDE Wallets to the Standby System

Copy the TDE wallet files from the primary DB system to standby DB system using SCP. The following sample command assumes the SCP command is being run by the oracle OS user and that the private key for oracle has been created and exists on the host where SCP is being run.

```
$ scp -i <private key> primary_server:/opt/oracle/dcs/commonstore/wallets/tde/<primary db_unique_name>/*
standby_server:/opt/oracle/dcs/commonstore/wallets/tde/<standby db_unique_name>
```

Setting Up the Standby System Configuration

As the oracle user, create the following directory for database version 11.2.0.4. This step is optional for version 12.2.0.1 and version 12.1.0.2.

```
[oracle@dbsys ~]$ mkdir  -pv /u03/app/oracle/redo/<standby db_unique_name uppercase>/controlfile
```

Creating the Audit File Destination

As the oracle user, create the following directory to use as the audit file destination.

```
[oracle@dbsys ~]$ mkdir -p /u01/app/oracle/admin/<db_name>/adump
```

Otherwise, the RMAN duplicate command used later will fail.

Creating a Temporary Password File

As the oracle user, create a temporary password file.

```
[oracle@dbsys ~]$ orapwd file=$ORACLE_HOME/dbs/orapw<standby oracle_sid>    password=<admin password for primary> entries=5
```

The password **must** be the same as the admin password of the primary database. Otherwise, the RMAN duplicate step below will fail with: RMAN-05614: Passwords for target and auxiliary connections must be the same when using active duplicate.

VERIFYING THE STANDBY DATABASE IS AVAILABLE

1. As the oracle user, set the environment variables.

   ```
   [oracle@dbsys ~]$ . oraenv
   <enter the db_name>
   ```

2. Replace $ORACLE_HOME/dbs/init*<standby sid_name>*.ora with the following content:

   ```
   db_name=<Primary db_name>
   db_unique_name=<standby db_unique_name>
   db_domain=<standby db_domain>
   ```

3. Remove the spfile from the standby.

   ```
   /u02/app/oracle/oradata/<standby db_unique_name>/dbs/spfile$ORACLE_SID.ora
   ```

   The database needs to be started in nomount mode with no spfile specified, but the original init file contains an spfile parameter which will prevent the RMAN duplicate step from working.

4. Set the ORACLE_UNQNAME environment variable to point to your DB_UNIQUE_NAME.

   ```
   $ export ORACLE_UNQNAME =db_unique_name
   ```

   Important

   If you do not perform this step, the wallet will not be opened, and running the RMAN DUPLICATE command in the subsequent step will fail.

5. The dbcli create-database --instanceonly command used earlier opens the

standby database as a primary in read/write mode, so the database needs to be brought down before proceeding to the nomount step below.

```
$ sqlplus / as sysdba
SQL> shutdown immediate
```

6. Start the database in nomount mode.

```
SQL> startup nomount
```

VERIFYING THE DATABASE CONNECTIONS

Verify the connection between the primary and standby databases.

1. Make sure that the listener port 1521 is open in the security list(s) used for the primary and standby DB systems. For more information, see Updating the Security List for the DB System.
2. From the primary database, connect to standby database.

```
$ sqlplus sys/<password>@<standby net service name> as sysdba
```

3. From standby database, connect to primary database.

```
$ sqlplus sys/<password>@<primary net service name> as sysdba
```

RUNNING THE RMAN DUPLICATE COMMAND

Run the RMAN DUPLICATE command on the standby DB system, as the oracle user.

If the primary database is large, you can allocate additional channels to improve performance. For a newly installed database, one channel typically runs the database duplication in a couple of minutes.

Make sure that there are no errors generated by the RMAN DUPLICATE command. If errors occur, restart the database using the `init.ora` file (not spfile) in case it is generated under `$ORACLE_HOME/dbs` as part of RMAN DUPLICATE.

In the following examples, use lowercase for the *<Standby db_unique_name>* unless otherwise specified.

For ACFS storage layout, run the following commands.

```
$ rman target sys/<password>@<primary alias>  auxiliary sys/<password>@<standby alias> log=rman.out

RMAN> run { allocate channel prim1 type disk;
            allocate auxiliary channel sby type disk;
            duplicate target database for standby from active database
            dorecover
            spfile
            parameter_value_convert '/<Primary db_unique_name>/','/<Standby db_unique_name>/','/<Primary
db_unique_name uppercase>/','/<Standby db_unique_name uppercase >/'
            set db_unique_name='<Standby db_unique_name>'
            set db_create_file_dest='/u02/app/oracle/oradata/<Standby db_unique_name>'
            set dg_broker_config_file1='/u02/app/oracle/oradata/<Standby db_unique_name>/dbs/dr1<Standby
db_unique_name>.dat'
            set dg_broker_config_file2='/u02/app/oracle/oradata/<Standby db_unique_name>/dbs/dr2<Standby
db_unique_name>.dat'
            set dispatchers ='(PROTOCOL=TCP) (SERVICE=<Standby db_unique_name>XDB)'
            set instance_name='<Standby db_unique_name>'
            ;
           }
```

For ASM storage layout, run the following commands.

```
$ rman target sys/<password>@<primary alias>  auxiliary sys/<password>@<standby alias> log=rman.out

RMAN> run { allocate channel prim1 type disk;
            allocate auxiliary channel sby type disk;
            duplicate target database for standby from active database
            dorecover
            spfile
            parameter_value_convert '/<Primary db_unique_name>/','/<Standby db_unique_name>/','/<Primary
db_unique_name uppercase>/','/<Standby db_unique_name uppercase>/'
            set db_unique_name='<Standby db_unique_name>'
            set dg_broker_config_file1='+DATA/<Standby db_unique_name>/dr1<Standby db_unique_name>.dat'
            set dg_broker_config_file2='+DATA/<Standby db_unique_name>/dr2<Standby db_unique_name>.dat'
            set dispatchers ='(PROTOCOL=TCP) (SERVICE=<Standby db_unique_name>XDB)'
            set instance_name='<Standby db_unique_name>'
            ;
          }
```

Enabling Database FLASHBACK

1. As a Data Guard best practice, enable flashback and set `db_flashback_retention_target` to at least 120 minutes on both the primary and standby databases.

```
SQL> alter database flashback on;
SQL> alter system set db_flashback_retention_target=120;
```

2. Verify that the standby database is created properly.

```
SQL> select FORCE_LOGGING, FLASHBACK_ON, OPEN_MODE, DATABASE_ROLE,SWITCHOVER_STATUS, DATAGUARD_
BROKER, PROTECTION_MODE from v$database ;
```

Creating a Database Service

Oracle recommends creating a database service for the standby database by using srvctl.

For ACFS storage layout.

1. Create a shared directory and copy the spfile file to it.

```
$ mkdir -pv /u02/app/oracle/oradata/<Standby db_unique_name>/dbs
$ cp $ORACLE_HOME/dbs/spfile<standby oracle_sid>.ora /u02/app/oracle/oradata/<Standby db_unique_
name>/dbs
```

2. Stop and remove the existing database service.

```
$ srvctl stop database -d <standby db_unique_name>
$ srvctl remove database -d <standby db_unique_name>
```

3. Create the database service.

```
$ srvctl add database -d <standby db_unique_name> -n <standby db_name> -o $ORACLE_HOME -c SINGLE
-p '/u02/app/oracle/oradata/<standby db_unique_name>/dbs/spfile<standby db_name>.ora'
-x <standby hostname> -s "READ ONLY" -r PHYSICAL_STANDBY -i <db_name>
$ srvctl setenv database -d <standby db_unique_name> -t "ORACLE_
UNQNAME=<standby db_unique_name>"
$ srvctl config database -d <standby db_unique_name>
```

4. Start the database service.

```
$ srvctl start database -d <standby db_unique_name>
```

5. Clean up the files from `$ORACLE_HOME/dbs`.

```
$ rm $ORACLE_HOME/dbs/spfile<standby oracle_sid>.ora
$ rm $ORACLE_HOME/dbs/init<standby oracle_sid>.ora
```

6. Create the `$ORACLE_HOME/dbs/init<standby oracle_sid>.ora` file to reference the new location of the spfile file.

```
SPFILE='/u02/app/oracle/oradata/<standby db_unique_name>/dbs/spfile<standby db_name>.ora'
```

7. Stop the standby database and then start it by using srvctl.

```
srvctl stop database -d <standby db_unique_name>
srvctl start database -d <standby db_unique_name>
```

For ASM storage layout.

1. Consider generating the spfile file under +DATA.

```
SQL> create pfile='init<standby oracle_sid>.ora' from spfile ;
SQL> create spfile='+DATA' from pfile='init<standby oracle_sid>.ora' ;
```

2. Stop and remove the existing database service.

```
$ srvctl stop database -d <standby db_unique_name>
$ srvctl remove database -d <standby db_unique_name>
```

3. Create the database service.

```
$ srvctl add database -d <standby db_unique_name> -n <standby db_name>   -o $ORACLE_HOME -c
SINGLE  -p '+DATA/<standby db_unique_name>/PARAMETERFILE/spfile.xxx.xxxxxx'
-x <standby hostname> -s "READ ONLY" -r PHYSICAL_STANDBY -i <db_name>
$ srvctl setenv database -d <standby db_unique_name> -t "ORACLE_UNQNAME=<standby db_unique_name>"
$ srvctl config database -d <standby db_unique_name>
```

4. Start the database service.

```
$ srvctl start database -d <standby db_unique_name>
```

5. Clean up the files from `$ORACLE_HOME/dbs`.

```
$ rm $ORACLE_HOME/dbs/init<standby oracle_sid>.ora
$ rm $ORACLE_HOME/dbs/spfile<standby oracle_sid>.ora
```

6. Create `$ORACLE_HOME/dbs/init<standby oracle_sid>.ora` file to reference the new location of the spfile file.

```
SPFILE='+DATA/<standby db_unique_name>/PARAMETERFILE/spfile.xxx.xxxxxx'
```

7. Stop the database and start the standby database by using srvctl.

```
$ srvctl start database -d <standby db_unique_name>
```

Configuring Data Guard

Perform the following steps to complete the configuration of Data Guard and enable redo transport from the primary database and redo apply in the standby database.

1. Run the dgmgrl command line utility from either the primary or standby DB system and connect to the primary database using sys credentials.

```
DGMGRL> connect sys/<sys password>@<primary tns alias>
```

2. Create the Data Guard configuration and identify for the primary and standby databases.

```
DGMGRL> create configuration mystby as primary database is <primary db_unique_name> connect
identifier is <primary tns alias>;
add database  <standby db_unique_name> as connect identifier is <standby tns alias>  maintained
as physical;
```

3. Enable Data Guard configuration.

```
DGMGRL> enable configuration;
```

4. Verify that Data Guard setup was done properly. Run the following SQL in **both** the primary and standby databases.

```
SQL> select FORCE_LOGGING, FLASHBACK_ON, OPEN_MODE, DATABASE_ROLE, SWITCHOVER_STATUS, DATAGUARD_
BROKER, PROTECTION_MODE from v$database;
```

5. Verify that Data Guard processes are initiated in the standby database.

```
SQL> select PROCESS,PID,DELAY_MINS from V$MANAGED_STANDBY;
```

6. Verify parameter configuration on primary and standby.

```
SQL> show parameter log_archive_dest_
SQL> show parameter log_archive_config
SQL> show parameter fal_server
SQL> show parameter log_archive_format
```

7. Verify that the Data Guard configuration is working. Specifically, make sure redo

shipping and redo apply are working and that the standby is not unreasonably lagging behind the primary.

```
DGMGRL> show configuration verbose
DGMGRL> show database verbose <standby db_unique_name>
DGMGRL> show database verbose <primary db_unique_name>
```

Any discrepancies, errors, or warnings should be resolved. You can also run a transaction on the primary and verify that it's visible in the standby.

8. Verify that the Data Guard configuration is functioning as expected by performing switchover and failover in both directions. Run `show configuration` after each operation and make sure there are no errors or warnings.



Warning

This step is optional, based on your discretion. If for any reason the configuration is not valid, the switchover and/or failover will fail and it might be difficult or impossible to start the primary database. A recovery of the primary might be required, which will affect availability.

```
DGMGRL> switchover to <standby db_unique_name>
DGMGRL> switchover to <primary db_unique_name>

#connect to standby before failover:

DGMGRL> connect sys/<sys password>@<standby db_unique_name>
DGMGRL> failover to <standby db_unique_name>
DGMGRL> reinstate database <primary db_unique_name>

#connect to primary before failover:

DGMGRL> connect sys/<sys password>@<primary db_unique_name>
DGMGRL> failover to <primary db_unique_name>
DGMGRL> reinstate database <standby db_unique_name>
```

Configuring Observer (Optional)

The best practice for high availability and durability is to run the primary, standby, and observer in separate availability domains. The observer determines whether or not to failover to a specific target standby database. The server used for observer requires the Oracle Client Administrator software, which includes the Oracle SQL NET and Broker.

1. Configure TNS alias names for both the primary and standby databases as described previously, and verify the connection to both databases.
2. Change protection mode to either maxavailability or maxperformance (maxprotection is not supported for FSFO).

   To enable maxavailability:

   ```
   DGMGRL> edit database <standby db_unique_name> set property 'logXptMode'='SYNC';
   DGMGRL> edit database <primary db_unique_name> set property 'logXptMode'='SYNC';
   DGMGRL> edit configuration set protection mode as maxavailability;
   ```

   To enable maxperformance:

   ```
   DGMGRL> edit configuration set protection mode as maxperformance;
   DGMGRL> edit database <standby db_unique_name> set property 'logXptMode'='ASYNC';
   DGMGRL> edit database <primary db_unique_name> set property 'logXptMode'='ASYNC';
   ```

   For maxperformance, the FastStartFailoverLaglimit property limits the maximum amount of permitted data loss to 30 seconds by default.
3. The following properties should also be considered. Run `show configuration verbose` to see their current values.
   - FastStartFailoverPmyShutdown
   - FastStartFailoverThreshold
   - FastStartFailoverTarget
   - FastStartFailoverAutoReinstate

   (Running `show configuration` will result in the following error until the observer is started: `Warning : ORA-16819: fast-start failover observer not started`.)
4. Enable fast-start failover from Broker:

   ```
   DGMGRL> Enable fast_start failover
   ```

5. Verify the fast-start failover and associated settings.

```
DGMGRL> show fast_start failover
```

6. Start the observer from Broker (it will run in the foreground, but can also be run in the background).

```
DGMGRL> start observer
```

7. Verify fast-start failover is enabled and without errors or warnings.

```
DGMGRL> show configuration verbose
```

8. Always test failover in both directions to ensure that everything is working as expected. Verify that FSFO is running properly by performing a shutdown abort of the primary database.

   The observer should start the failover to the standby database. If protection mode is set to maxprotection, some loss of data can occur, based on the FastStartFailoverLaglimit value.

Oracle Database CLI Reference

The database CLI (dbcli) is a command line interface available on bare metal and virtual machine DB systems. After you connect to the DB system, you can use the database CLI to perform tasks such as creating Oracle database homes and databases.



Note

The database CLI is **not** for use on Exadata DB systems.

Operational Notes

- The database CLI commands must be run as the root user.
- `dbcli` is in the `/opt/oracle/dcs/bin/` directory.

  This directory is included in the path for the root user's environment.

- Oracle Database maintains logs of the `dbcli` command output in the `dcscli.log` and `dcs-agent.log` files in the `/opt/oracle/dcs/log/` directory.
- The database CLI commands and most parameters are case sensitive and should be typed as shown. A few parameters are not case sensitive, as indicated in the parameter descriptions, and can be typed in uppercase or lowercase.



Warning

Oracle recommends that you avoid specifying parameter values that include confidential information when you use the database CLI commands.

Syntax

The database CLI commands use the following syntax:

```
dbcli command [parameters]
```

where:

- `command` is a verb-object combination such as `create-database`.
- `parameters` include additional options for the command. Most parameter names are preceded with two dashes, for example, `--help`. Abbreviated parameter names are preceded with one dash, for example, `-h`.
- User-specified parameter values are shown in red text within angle brackets, for example, *<db_home_id>*. Omit the angle brackets when specifying these values.
- The help parameter is available with every command.

The remainder of this topic contains syntax and other details about the commands.

CLI Update Command

Occasionally, new commands are added to the database CLI and other commands are updated to support new features. You can use the following command to update the database CLI:

CLIADM UPDATE-DBCLI

Use the `cliadm update-dbcli` command to update the database CLI with the latest new and updated commands.



Note

The `cliadm update-dbcli` command is not available on 2-node RAC DB systems.

SYNTAX

```
cliadm update-dbcli [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.

EXAMPLE

The following command updates the `dbcli`:

```
[root@dbsys ~]# cliadm update-dbcli
{
  "jobId" : "dc9ce73d-ed71-4473-99cd-9663b9d79bfd",
  "status" : "Created",
  "message" : "Dcs cli will be updated",
  "reports" : [ ],
  "createTimestamp" : "January 18, 2017 10:19:34 AM PST",
  "resourceList" : [ ],
  "description" : "dbcli patching",
  "updatedTime" : "January 18, 2017 10:19:34 AM PST"
}
```

Agent Commands

The following commands are available to manage agents:

- dbcli ping-agent
- dbcli list-agentConfigParameters
- dbcli update-agentConfigParameters

DBCLI PING-AGENT

Use the `dbcli ping-agent` command to test the reachability of an agent.

SYNTAX

```
dbcli ping-agent [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.

DBCLI LIST-AGENTCONFIGPARAMETERS

Use the `dbcli list-agentConfigParameters` command to list agent configuration parameters.

SYNTAX

```
dbcli list-agentConfigParameters [-n] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.
-n	-name	(Optional) Parameter name.

DBCLI UPDATE-AGENTCONFIGPARAMETERS

Use the `dbcli update-agentConfigParameters` command to update agent configuration parameters.

SYNTAX

```
dbcli update-agentConfigParameters -n <parameter> [-v <value>] [-a] [-c] [-d] [-u] [-r] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-a	--append	(Optional) Appends the specified values to the specified parameters. Example with multiple parameter names and values: `-n p1 -v v1 -n p2 -v v2 -a`
-c	--comment	(Optional) Adds a comment for the parameter. Default: [ ]
-d	--description	(Optional) Adds a description for the parameter. Default: [ ]
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.
-n	--name	Parameter name. Example with multiple parameter names and values: `-n p1 -v v1 -n p2 -v v2` Default: [ ]

Parameter	Full Name	Description
-r	--reset	(Optional) Resets the parameter to the default value. Example resetting multiple parameters: `-n p1 -n p2 -r` Default: false
-u	--update	(Optional) Replaces the specified parameter values as directed. Example with multiple parameter names and values: `-n p1 -v v1 -n p2 -v v2 -u` Default: false
-v	--value	(Optional) Parameter value. Example with multiple parameter names and values: `-n p1 -v v1 -n p2 -v v2` Default: [ ]

Autologcleanpolicy Commands

The following commands are available to manage policies for automatic cleaning (purging) of logs.

- dbcli create-autoLogCleanPolicy
- dbcli list-autoLogCleanPolicy

DBCLI CREATE-AUTOLOGCLEANPOLICY

Use the `dbcli create-autoLogCleanPolicy` command to create policies for automatic cleaning (purging) of logs.

SYNTAX

```
dbcli create-autoLogCleanPolicy [-c {gi|database|dcs}] [-f <number>] [-o <number>] [-u
{Day|Hour|Minute}] [-uMB <number>] [-uPer <number>] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-c	--components	(Optional) Components to purge. Possible values are gi, database, and dcs. Separate multiple values with commas. Example: `gi,dcs`
-f	--freeSpaceBelowPercentage	(Optional) Purges logs when the free disk space is below the specified percentage of the total partition size. Valid range: 20-50. Default: 20.
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.
-o	--olderthan	(Optional) Quantity portion of time interval. Default: 30. Cleans logs older than the specified time interval (`-o` and `-u`).
-u	--olderThanUnit	(Optional) Unit portion of time interval. Possible values: Day, Hour, or Minute. Default: Day. Cleans logs older than the specified time interval (`-o` and `-u`).
-uMB	--usageOverMB	(Optional) Purges logs when log usage exceeds the specified number of MegaBytes (MB). Valid range: 10 to 50% of total partition size.
-uPer	--usageOverPercentage	(Optional) Purges logs when log usage exceeds the specified percentage of the total partition size. Valid range: 10-50.

DBCLI LIST-AUTOLOGCLEANPOLICY

Use the `dbcli list-autoLogCleanPolicy` command to list policies for automatic cleaning of logs.

SYNTAX

```
dbcli list-autoLogCleanPolicy [-c {gi|database|dcs}] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-c	--components	(Optional) Components. Possible values are gi, database, and dcs. Separate multiple values with commas. Example: `gi,dcs`
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.

Backup Commands

The following commands are available to back up databases:

- dbcli create-backup
- dbcli getstatus-backup
- dbcli schedule-backup



Note

Instead of using `dbcli`, you can use the Console or the API to manage backing up your bare metal or virtual machine DB system databases to Object Storage. However, if you switch from using `dbcli` to using managed backups, a new backup configuration is created and associated with your database, and backups you created by using `dbcli` will not be accessible from the managed backup interfaces. For information about managed backups, see Backing Up to Oracle Cloud Infrastructure Object Storage.

Before you can back up a database by using the dbcli create-backup command, you'll need to:

1. Create a backup configuration by using the dbcli create-backupconfig command.
2. Associate the backup configuration with the database by using the dbcli update-database command.

After a database is associated with a backup configuration, you can use the `dbcli create-backup` command in a `cron` job to run backups automatically. You can use a cron utility such as CronMaker to help build expressions. For more information, see http://www.cronmaker.com.

DBCLI CREATE-BACKUP

Use the `dbcli create-backup` command to create a backup of a database.

SYNTAX

```
dbcli create-backup -in <db_name> -i <db_id> [-bt {Regular-L0|Regular-L1|Longterm|ArchiveLog}] [-c
{Database|TdeWallet}] [-k <n>] [-t <tag>] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-bt	--backupType	(Optional) Backup type. Possible values are Regular-L0, Regular-L1, Longterm, and ArchiveLog. Regular-L0 and Regular L1 correspond to incremental L0 and L1 backups. Longterm corresponds to Full backup. ArchiveLog corresponds to archived redo logs backup. The default value is Regular-L1. Values are not case-sensitive. If omitted, the default value is used.

Parameter	Full Name	Description
-c	--component	(Optional) Component. Possible values are Database and TdeWallet. The default value is Database. The value TdeWallet backs up TDE wallets. Values are not case-sensitive. If omitted, the default value is used. **Note** TDE wallets are automatically backed up in the following situations: • A database is created with an Object Storage backup configuration. • A database that has an Object Storage backup configuration is updated. • An Object Storage backup configuration is updated. • A backup of the type Longterm is created. • The TDE key for a database is rotated. • A database is backed up and no TDE wallet backups exist yet.
-h	--help	(Optional) Displays help for using the command.

Parameter	Full Name	Description
-i	--dbid	The ID of the database to back up. Use the `dbcli list-databases` command to get the database's ID.
-in	--dbName	The name of the database to back up. Use the `dbcli list-databases` command to get the database's name.
-j	--json	(Optional) Displays JSON output.
-k	--keepDays	(Optional) Specifies the time until which the backup or copy must be kept. After this time the backup is obsolete, regardless of the backup retention policy settings. For Longterm backup type only.
-t	--tag	(Required for Longterm backup type) Specifies a user-specified tag name for a backup set and applies this tag to the output files generated by the command. This value is not case sensitive. Valid number of characters: 1 to 30. The characters are limited to the characters that are valid in file names on the target file system. For example, ASM does not support the use of the hyphen (-) character in the file names it uses internally, so weekly-incremental is not a valid tag name for backups in ASM disk groups. Environment variables are not valid in the TAG parameter.

EXAMPLES

The following command creates a backup of the specified database using the database ID.

```
[root@dbsys ~]# dbcli create-backup -i 573cadb2-0cc2-4c1c-9c31-595ab8963d5b
```

The following command creates a backup of the specified database using the database name ("mydb").

```
[root@dbsys ~]# dbcli create-backup -in mydb
```

DBCLI GETSTATUS-BACKUP

Use the `dbcli getstatus-backup` command to display the status of a backup.

SYNTAX

```
dbcli getstatus-backup -t <backup_type> [i <id>] [-in <name>] [-l] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-i	--dbId	(Optional) Database Resource ID.
-in	--dbName	(Optional) Database Resource Name.
-j	--json	(Optional) Displays JSON output.
-l	--isLatestBackupReport	(Optional) Latest backup report. Default: true.
-t	--backupType	Backup type.

DBCLI SCHEDULE-BACKUP

Use the `dbcli schedule-backup` command to schedule a backup of a database.

SYNTAX

```
dbcli schedule-backup -t <backup_type> -f <number> [i <id>] [-in <name>] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-f	--frequency	Frequency in minutes.
-h	--help	(Optional) Displays help for using the command.

Parameter	Full Name	Description
-i	--dbId	(Optional) Database Resource ID.
-in	--dbName	(Optional) Database Resource Name.
-j	--json	(Optional) Displays JSON output.
-t	--backupType	Backup type.

Backupconfig Commands

A backup configuration determines the backup destination and recovery window for database backups. You create the backup configuration and then associate it with a database by using the `dbcli update-database` command.



Warning

Backups that were configured using the Console may become unusable if you make changes using these commands. For backups configured using the Console, use these commands with support guidance only.



Note

Instead of using `dbcli`, you can use the Console or the API to manage backing up your bare metal or virtual machine DB system databases to Object Storage. For information about managed backups, see Backing Up to Oracle Cloud Infrastructure Object Storage.

After a database is associated with a backup configuration, you can use the `dbcli create-backup` command in a `cron` job to run backups automatically. You can use a cron utility such

as CronMaker to help build expressions. For more information, see http://www.cronmaker.com.

The following commands are available to manage backup configurations:

- dbcli create-backupconfig
- dbcli list-backupconfigs
- dbcli describe-backupconfig
- dbcli update-backupconfig
- dbcli delete-backupconfig

DBCLI CREATE-BACKUPCONFIG

Use the `dbcli create-backupconfig` command to create a backup configuration that defines the backup destination and recovery windows.

SYNTAX

```
dbcli create-backupconfig -d {DISK|OBJECTSTORE|NONE} -c <bucket>  -o <object_store_swift_id> -on
<object_store_swift_name> -w <n> -n <name> [-cr|-no-cr] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-c	--container	The name of an existing bucket in the Oracle Cloud Infrastructure Object Storage service. You can use the Console or the Object Storage API to create the bucket. For more information, see Managing Buckets. You must also specify `--backupdestination objectstore` and the `--objectstoreswiftId` parameter.
-cr -no-cr	--crosscheck --no-crosscheck	(Optional) Indicates whether to enable the crosscheck operation. This operation determines if the files on the disk or in the media management catalog correspond to data in the RMAN repository. If omitted, the default setting is used (crosscheck is enabled by default).
-d	--backupdestination	The backup destination as one of the following (these values are **not** case sensitive): DISK - The local Fast Recovery Area. OBJECTSTORE - The Oracle Cloud Infrastructure Object Storage service. You must also specify the `--container` and `--objectstoreswiftId` parameters. NONE - Disables the backup.
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.
-n	--name	The name of the backup configuration.

Parameter	Full Name	Description
-o	--objectstoreswiftId	The ID of the object store that contains the endpoint and credentials for the Oracle Cloud Infrastructure Object Storage service. Use the dbcli list-objectstoreswifts command to get the object store ID. Use the dbcli create-objectstoreswift command to create an object store. You must also specify `--backupdestination objectstore` and the `--container` parameter.
-on	--objectstoreswiftName	The name of the object store that contains the endpoint and credentials for the Oracle Cloud Infrastructure Object Storage service. Use the dbcli list-objectstoreswifts command to get the object store name. Use the dbcli create-objectstoreswift command to create an object store. You must also specify `--backupdestination objectstore` and the `--container` parameter.
-w	--recoverywindow	The number of days for which backups and archived redo logs are maintained. The interval always ends with the current time and extends back in time for the number of days specified. For a DISK backup destination, specify 1 to 14 days. For an OBJECTSTORE backup destination, specify 1 to 30 days.

EXAMPLE

The following command creates a backup configuration named dbbkcfg1:

```
[root@dbsys ~]# dbcli create-backupconfig -d Disk -w 7 -n dbbkcfg1
{
```

```
  "jobId" : "4e0e6011-db53-4142-82ef-eb561658a0a9",
  "status" : "Success",
  "message" : null,
  "reports" : [ {
    "taskId" : "TaskParallel_919",
    "taskName" : "persisting backup config metadata",
    "taskResult" : "Success",
    "startTime" : "November 18, 2016 20:21:25 PM UTC",
    "endTime" : "November 18, 2016 20:21:25 PM UTC",
    "status" : "Success",
    "taskDescription" : null,
    "parentTaskId" : "TaskSequential_915",
    "jobId" : "4e0e6011-db53-4142-82ef-eb561658a0a9",
    "tags" : [ ],
    "reportLevel" : "Info",
    "updatedTime" : "November 18, 2016 20:21:25 PM UTC"
  } ],
  "createTimestamp" : "November 18, 2016 20:21:25 PM UTC",
  "description" : "create backup config:dbbkcfg1",
  "updatedTime" : "November 18, 2016 20:21:25 PM UTC"
}
```


DBCLI LIST-BACKUPCONFIGS

Use the `dbcli list-backupconfigs` command to list all the backup configurations in the DB system.

SYNTAX

```
dbcli list-backupconfigs [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.

EXAMPLE

The following command lists a backup configuration:

```
[root@dbsys ~]# dbcli list-backupconfigs


ID                                       Name                 RecoveryWindow     BackupDestination
CreateTime
---------------------------------------- -------------------- ------------------ ----------------- -----
------------------------
ccdd56fe-a40b-4e82-b38d-5f76c265282d     dbbkcfg1             7                  Disk              July
10, 2016 12:24:08 PM UTC
```

DBCLI DESCRIBE-BACKUPCONFIG

Use the `dbcli describe-backupconfig` command to show details about a specific backup configuration.

SYNTAX

```
dbcli describe-backupconfig -i <id> -in <name> [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-i	--backupconfigid	The backup configuration ID. Use the `dbcli list-backupconfigs` command to get the ID.
-in	--backupconfigname	The backup configuration name. Use the `dbcli list-backupconfigs` command to get the name.
-j	--json	(Optional) Displays JSON output.

EXAMPLE

The following command displays details about a backup configuration:

```
[root@dbsys ~]# dbcli describe-backupconfig -i ccdd56fe-a40b-4e82-b38d-5f76c265282d

Backup Config details
----------------------------------------------------------------
```

```
                    ID: ccdd56fe-a40b-4e82-b38d-5f76c265282d
                  Name: dbbkcfg1
        RecoveryWindow: 7
     BackupDestination: Disk
           CreatedTime: July 10, 2016 12:24:08 PM UTC
           UpdatedTime: July 10, 2016 12:24:08 PM UTC
```

DBCLI UPDATE-BACKUPCONFIG

Use the `dbcli update-backupconfig` command to update an existing backup configuration.

SYNTAX

```
dbcli update-backupconfig -i <id> -in <name> -w <n> -c <bucket> -o <object_store_swift_id> -on <object_store_swift_name> [-cr|-no-cr] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-c	--container	The name of an existing bucket in the Oracle Cloud Infrastructure Object Storage service. You can use the Console or the Object Storage API to create the bucket. For more information, see Managing Buckets. You must also specify `--backupdestination objectstore` and the `--objectstoreswiftId` parameter.
-cr -no-cr	--crosscheck --no-crosscheck	(Optional) Indicates whether to enable the crosscheck operation. This operation determines if the files on the disk on in the media management catalog correspond to data in the RMAN repository. If omitted, the default setting is used (crosscheck is enabled by default).
-h	--help	(Optional) Displays help for using the command.

Parameter	Full Name	Description
-i	--backupconfigid	The ID of the backup configuration to update. Use the `dbcli list-backupconfigs` command to get the ID.
-in	--backupconfigname	The name of the backup configuration to update. Use the `dbcli list-backupconfigs` command to get the name.
-j	--json	(Optional) Displays JSON output.
-o	--objectstoreswiftId	The ID of the object store that contains the endpoint and credentials for the Oracle Cloud Infrastructure Object Storage service. Use the dbcli list-objectstoreswifts command to get the object store ID. Use the dbcli create-objectstoreswift command to create an object store. You must also specify `--backupdestination objectstore` and the `--container` parameter.
-on	--objectstoreswiftname	The name of the object store that contains the endpoint and credentials for the Oracle Cloud Infrastructure Object Storage service. Use the dbcli list-objectstoreswifts command to get the object store name. Use the dbcli create-objectstoreswift command to create an object store. You must also specify `--backupdestination objectstore` and the `--container` parameter.
-w	--recoverywindow	The new disk recovery window. For a DISK backup destination, specify 1 to 14 days. For an OBJECTSTORE backup destination, specify 1 to 30 days.

EXAMPLE

The following command updates the recovery window for a backup configuration:

```
[root@dbsys ~]# dbcli update-backupconfig -i ccdd56fe-a40b-4e82-b38d-5f76c265282d -w 5
{
  "jobId" : "0e849291-e1e1-4c7a-8dd2-62b522b9b807",
  "status" : "Created",
  "message" : null,
  "reports" : [ ],
  "createTimestamp" : 1468153731699,
  "description" : "update backup config: dbbkcfg1",
  "updatedTime" : 1468153731700
}
```

DBCLI DELETE-BACKUPCONFIG

Use the `dbcli delete-backupconfig` command to delete a backup configuration.

SYNTAX

```
dbcli delete-backupconfig -i <id> -in <name> [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-i	--id	The backup configuration ID to delete. Use the `dbcli list-backupconfigs` command to get the ID.
-in	--backupconfigname	The name of the backup configuration to delete. Use the `dbcli list-backupconfigs` command to get the name.
-j	--json	(Optional) Displays JSON output.

EXAMPLE

The following command deletes the specified backup configuration:

```
[root@dbsys ~]# dbcli delete-backupconfig -i ccdd56fe-a40b-4e82-b38d-5f76c265282d
```

Bmccredential Commands

The following commands are available to manage credentials configurations, which are required for downloading DB system patches from the Oracle Cloud Infrastructure Object Storage service. For more information, see Patching a DB System.

- dbcli create-bmccredential
- dbcli list-bmccredentials
- dbcli describe-bmccredential
- dbcli delete-bmccredential
- dbcli update-bmccredential



Note

The `bmccredential` commands are not available on 2-node RAC DB systems.

DBCLI CREATE-BMCCREDENTIAL

Use the `dbcli create-bmccredential` command to create a credentials configuration.

PREREQUISITES

Before you can create a credentials configuration, you'll need these items:

- An RSA key pair **in PEM format** (minimum 2048 bits). See How to Generate an API Signing Key.
- The fingerprint of the public key. See How to Get the Key's Fingerprint.
- Your tenancy's OCID and user name's OCID. See Where to Get the Tenancy's OCID and User's OCID.

Then you'll need to upload the public key in the Console. See How to Upload the Public Key.

SYNTAX

```
dbcli create-bmccredential -c [backup|patching|other] -t <tenant_ocid> -u <user_ocid>  -f <fingerprint>
-k <private_key_path> -p|-hp <passphrase> [-n <credentials_name>] [-e <object_store_url>] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-c	--credentialsType	The type of Object Storage credentials configuration to create (these values are **not** case sensitive): BACKUP - Reserved for the future use. PATCHING - For downloading patches from the service. OTHER - Reserved for the future use.
-e	--objectStoreUrl	(Optional) The Object Storage endpoint URL. Omit this parameter when `--credentialsType PATCHING` is specified. The following URL is assumed: https://objectstorage.*<region_name>*.oraclecloud.com See Regions and Availability Domains for region name strings.
-f	--fingerPrint	The public key fingerprint. You can find the fingerprint in the Console by clicking your user name in the upper right corner and then clicking **User Settings**. The fingerprint looks something like this: `-f 61:9e:52:26:4b:dd:46:dc:8c:a8:05:6b:9f:0a:30:d2`
-k	--privateKey	The path to the private key file in PEM format, for example: `-k /root/.ssh/privkey`
-h	--help	(Optional) Displays help for using the command.

Parameter	Full Name	Description
-j	--json	(Optional) Displays JSON output.
-n	--name	(Optional) The name for the new credentials configuration. The name is useful for tracking the configuration.
-p -hp	--passPhrase	The passphrase for the public/private key pair, if you specified one when creating the key pair. Specify `-p` (with no passphrase) to be prompted. Specify `-hp` *<passphrase>* to provide the passphrase in the command.
-t	--tenantOcid	Your tenancy OCID. See Where to Find Your Tenancy's OCID. The tenancy OCID looks something like this: `ocid1.tenancy.oc1..<unique_ID>`
-u	--userOcid	The user name OCID for your Oracle Cloud Infrastructure user account. You can find the OCID in the Console: Open the **User** menu () and click **User Settings**. The user name OCID looks something like this: `ocid1.user.oc1..<unique_ID>`

EXAMPLE

The following command creates a credentials configuration:

```
[root@dbsys ~]# dbcli create-bmccredential -c patching -hp mypass -t
ocid1.tenancy.oc1..aaaaaaaaba3pv6wkcr4jqae5f44n2b2m2yt2j6rx32uzr4h25vqstifsfdsq -u
ocid1.user.oc1..aaaaaaaalhdxviuxqi7xevqskscc16edokgldvuf6raskcioq4x2z7watsfa -f
60:9e:56:26:4b:dd:46:dc:8c:a8:05:6d:9f:0a:30:d2 -k /root/.ssh/privkey

{
  "jobId" : "f8c80510-b717-4ee2-a47e-cd380480b28b",
  "status" : "Created",
  "message" : null,
  "reports" : [ ],
```

```
  "createTimestamp" : "December 26, 2016 22:46:38 PM PST",
  "resourceList" : [ ],
  "description" : "BMC Credentials Creation",
  "updatedTime" : "December 26, 2016 22:46:38 PM PST"
}
```

DBCLI LIST-BMCCREDENTIALS

Use the `dbcli list-bmccredentials` command to list the credentials configurations on the DB system.

SYNTAX

```
dbcli list-bmccredentials [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.

EXAMPLE

The following command lists the credentials configurations on the DB system:

```
[root@dbsys ~]# dbcli list-bmccredentials
    ID                                       Name           Type          End Point
                              Status
    ---------------------------------------- -------------  ----------    ------------------------------
----------------------------- ----------
    f19d7c8b-d0d5-4jhf-852b-eb2a81cb7ce5     patch1         Patching      https://objectstorage.us-
phoenix-1.oraclecloud.com           Configured
    f1a8741c-b0c4-4jhf-239b-ab2a81jhfde4     patch2         Patching      https://objectstorage.us-
phoenix-1.oraclecloud.com           Configured
```

DBCLI DESCRIBE-BMCCREDENTIAL

Use the `dbcli describe-bmccredential` command to display details about a credentials configuration.

SYNTAX

```
dbcli describe-bmccredential -i <credentials_id> [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-i	--id	The ID for the credentials configuration. Use the dbcli list-bmccredentials command to get the ID.
-j	--json	(Optional) Displays JSON output.

EXAMPLE

The following command displays details about the specified credentials configuration:

```
[root@dbsys ~]# dbcli describe-bmccredential -i 09f9988e-eed5-4dde-8814-890828d1c763

BMC Credentials details
----------------------------------------------------------------
                     ID: 09f9988e-eed5-4dde-8814-890678d1c763
                   Name: patch23
           Tenant OCID: ocid1.tenancy.oc1..aaaaaaaaba3pv6wkcr4jqae5f44n2b2m2yt2j6rx32uzr4h25vqstifsfdsq
             User OCID: ocid1.user.oc1..aaaaaaaalhjhfiuxqi7xevqskscc16edokgldvuf6raskcioq4x2z7watjhf
      Credentials Type: Patching
       objectStore URL: https://objectstorage.us-phoenix-1.oraclecloud.com
                Status: Configured
               Created: January 9, 2017 1:19:11 AM PST
           UpdatedTime: January 9, 2017 1:41:46 AM PST
```

DBCLI DELETE-BMCCREDENTIAL

Use the `dbcli delete-bmccredential` command to delete a credentials configuration.

SYNTAX

```
dbcli delete-bmccredential -i <credentials_id> [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-i	--id	The ID for the credentials configuration. Use the dbcli list-bmccredentials command to get the ID.
-j	--json	(Optional) Displays JSON output.

EXAMPLE

The following command deletes the specified credentials configuration:

```
[root@dbsys ~]# dbcli delete-bmccredential -i f19d7c8b-d0d5-4jhf-852b-eb2a81cb7ce5
```

DBCLI UPDATE-BMCCREDENTIAL

Use the `dbcli update-bmccredential` command to update a credentials configuration.

SYNTAX

```
dbcli update-bmccredential -i <credentials_id> -n <credentials_name> -c [backup|patching|other] -p|-hp
<passphrase> -f <fingerprint> -k <private_key_path> [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-c	--credentialsType	The type of Object Storage credentials configuration (these values are **not** case sensitive): BACKUP - Reserved for the future use. PATCHING - For downloading patches from the service. OTHER - Reserved for the future use.
-i	--id	The ID for the credentials configuration. Use the dbcli list-bmccredentials command to get the ID.
-f	--fingerPrint	The public key fingerprint, for example: `-f 61:9e:52:26:4b:dd:46:dc:8c:a8:05:6b:9f:0a:30:d2`
-k	--privateKey	The path to the private key file in PEM format, for example: `-k /root/.ssh/privkey`
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.
-n	--name	(Optional) The name for the credentials configuration. Use the dbcli list-bmccredentials command to get the name.
-p -hp	--passPhrase	The passphrase for the public/private key pair, if you specified one when creating the key pair. Specify `-p` (with no passphrase) to be prompted. Specify `-hp` *<passphrase>* to provide the passphrase in the command.

EXAMPLE

The following command updates a credentials configuration:

```
[root@dbsys ~]# dbcli update-bmccredential -c OTHER -i 6f921b29-61b6-56f4-889a-ce9270621956
{
  "jobId" : "6e95a69e-cf73-4e51-a444-c7e4b9631c27",
  "status" : "Created",
  "message" : null,
  "reports" : [ ],
  "createTimestamp" : "January 19, 2017 12:01:10 PM PST",
  "resourceList" : [ ],
  "description" : "Update BMC Credentials of object 6f921b29-61b6-48f4-889a-ce9270621945",
  "updatedTime" : "January 19, 2017 12:01:10 PM PST"
```

Component Command

DBCLI DESCRIBE-COMPONENT



Tip

Your DB system might not include this newer command. If you have trouble running the command, use the cliadm update-dbcli command to update the database CLI and then retry the command.



Note

The `dbcli describe-component` command is not available on 2-node RAC DB systems. Patching 2-node systems from Object Storage is not supported.

Use the `dbcli describe-component` command to show the installed and available patch versions for the server, storage, and/or database home components in the DB system.

This command requires a valid Object Storage credentials configuration. Use the dbcli create-bmccredential command to create the configuration if you haven't already done so. If the

configuration is missing or invalid, the command fails with the error: `Failed to connect to the object store. Please provide valid details.`

For more information about updating the CLI, creating the credentials configuration, and applying patches, see Patching a DB System.

SYNTAX

```
dbcli describe-component [-s <server_group>] [-d <db_group>] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-d	--dbhomes	(Optional) Lists the installed and available patch versions for only the database home components.
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.
-s	--server	(Optional) Lists the installed and available patch versions for only the server components.

EXAMPLE

The following command to show the current component versions and the available patch versions in the object store:

```
[root@dbsys ~]# dbcli describe-component
System Version
---------------
12.1.2.10.0

Component                                Installed Version    Available Version
---------------------------------------- -------------------- --------------------
OAK                                      12.1.2.10.0          up-to-date
GI                                       12.1.0.2.161018      up-to-date
ORADB12102_HOME1                         12.1.0.2.161018      up-to-date
ORADB12102_HOME2, ORADB12102_HOME3       12.1.0.2.160719      12.1.0.2.161018
```

Database Commands

The following commands are available to manage databases:

- dbcli clone-database
- dbcli create-database
- dbcli delete-database
- dbcli describe-database
- dbcli list-databases
- dbcli modify-database
- dbcli recover-database
- dbcli register-database
- dbcli update-database

DBCLI CLONE-DATABASE

Use the `dbcli clone-database` command to clone a database.

SYNTAX

```
dbcli clone-database -f <name> -u <name> -n <name> [-s <shape>] [-t <type>] [m <sys_password>] [-p <tde_password>] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-f	--sourcedbname	Source database name.
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.
-m	--syspassword	(Optional) Password for SYS.
-n	--dbname	Database name.

Parameter	Full Name	Description
-p	--tdepassword	(Optional) Password for source TDE wallet.
-s	--dbshape	(Optional) Database shape. Examples: odb1, odb2.
-t	--dbtype	(Optional) Database Type: SI
-u	--databaseUniqueName	Database unique name.

DBCLI CREATE-DATABASE

Use the `dbcli create-database` command to create a new database. You can create a database with a new or existing Oracle Database home, however each database home can have only one database.

It takes a few minutes to create the database. After you run the `dbcli create-database` command, you can use the `dbcli list-jobs` command to check the status of the database creation job.



Tip

Wait for the database creation job to complete before you attempt to create another database. Running multiple `dbcli create-database` commands at the same time can result in some of the creation jobs not completing.

Once the database is created, you can use the `dbcli list-databases -j` command to see additional information about the database.



Note

The `dbcli create-database` command is available on bare metal DB systems only.

You must create and activate a master encryption key for any PDBs that you create. After creating or plugging in a new PDB on a 1- or 2-node RAC DB System, use the `dbcli update-tdekey` command to create and activate a master encryption key for the PDB. Otherwise, you might encounter the error `ORA-28374: typed master key not found in wallet` when attempting to create tablespaces in the PDB. In a multitenant environment, each PDB has its own master encryption key which is stored in a single keystore used by all containers. For more information, see "Overview of Managing a Multitenant Environment" in the Oracle Database Administrator's Guide.

SYNTAX

```
dbcli create-database -dh <db_home_id> -cl {OLTP|DSS|IMDB} -n <db_name> -u <unique_name> -bi <bkup_
config_id> -bn <bkup_config_name> -m -s <db_shape> -r {ACFS|ASM} -y {SI|RAC|RACOne} [-dn <name>] -io -d
<pdb_admin_user> [-p <pdb>] [-ns <nlcharset>] [-cs <charset>] [-l <language>] [-dt <territory>] -v
<version> [-c|-no-c] [-co|-no-co] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-bi	--backupconfigid	Defines the backup configuration identifier for future use. Use the `dbcli list-backupconfigs` command to get the ID.
-bn	--backupconfigname	Defines the backup configuration name for future use. Use the `dbcli list-backupconfigs` command to get the name.
-c -no-c	--cdb --no-cdb	(Optional) Indicates whether to create a Container Database. If omitted, a Container Database is not created.
-cs	--characterset	(Optional) Defines the character set for the database. The default is AL32UTF8.
-cl	--dbclass	Defines the database class. The options are OLTP, DSS, or IMDB. The default is OLTP. For Enterprise Editions, all three classes are supported. For Standard Edition, only OLTP is supported.
-co -no-co	--dbconsole --no-dbconsole	(Optional) Indicates whether the Database Console is enabled. If omitted, the console is not enabled. This parameter is not available for a version 11.2.0.4 database on a 2-node RAC DB system. For more information, see To enable the console for a version 11.2.0.4 database on a multi-node DB system .
-d	--pdbadmin	Defines the name of the Pluggable Database (PDB) Admin User. The default value is pdbadmin.

Parameter	Full Name	Description
-dn	--dbdomainname	(Optional) Database domain name (indicates the logical location of the database within the network structure).
-dt	--dbterritory	(Optional) Defines the territory for the database. The default is AMERICA.
-dh	--dbhomeid	Identifies the database home in which to create the database. The database home must be empty because each database home can have only one database. You can use the `dbcli list dbhomes` command to get the DB home ID. If this parameter is omitted, the database is created with a new Oracle home.
-h	--help	(Optional) Displays help for the command.
-j	--json	(Optional) Displays JSON output.
-l	--dblanguage	(Optional) Defines the language for the database. The default is AMERICAN.
-m	--adminpassword	A strong password for SYS, SYSTEM, TDE wallet, and PDB Admin. The password must be 9 to 30 characters and contain at least 2 uppercase, 2 lowercase, 2 numeric, and 2 special characters. The special characters must be _, #, or -. The password must not contain the username (SYS, SYSTEM, and so on) or the word "oracle" either in forward or reversed order and regardless of casing. Specify `-m` (with no password) to be prompted for the password.

Parameter	Full Name	Description
-n	--dbname	Defines the name given to the new database. The database name must begin with an alphabetic character and can contain a maximum of eight alphanumeric characters. Special characters are not permitted.
-ns	--nationalscharacterset	(Optional) Defines the national character set for the database. The default is AL16UTF16.
-p	--pdbname	(Optional) Defines a unique name for the PDB. The PDB name must begin with an alphabetic character and can contain a maximum of 30 alphanumeric characters. The only special character permitted is the underscore (_). The default value is pdb1. PDB names must be unique within a CDB and within the listener to which they are registered. Make sure the PDB name is unique on the system. To ensure uniqueness, do not use the default name value (pdb1).
-r	--dbstorage	Defines the database storage, either ACFS or ASM. The default value is ASM. See Usage Notes for more information.
-s	--dbshape	Identifies the database sizing template to use for the database. For example, odb1, odb2, or odb3. The default is odb1. For more information, see Database Sizing Templates.

Parameter	Full Name	Description
-u	--databaseUniqueName	Defines a unique name for the database to ensure uniqueness within an Oracle Data Guard group (a primary database and its standby databases). The unique name can contain only alphanumeric and underscore (_) characters. The unique name cannot be changed. The unique name defaults to the name specified in the `--dbname` parameter.
-v	--version	Defines the database version as one of the following: • 18.1.0.0 • 12.2.0.1 • 12.1.0.2 (the default) • 11.2.0.4
-y	--dbtype	Defines the database type. Specify SI for a 1-node instance, RAC for a 2-node cluster, or RACOne for 1-node instance with a second node in cold standby mode. The default value is RAC. These values are not case sensitive.

USAGE NOTES

- You cannot mix Oracle Database Standard Edition and Enterprise Edition databases on the same DB system. (You can mix supported database versions on the DB system, but not editions.)
- When `--dbhomeid` is not provided, the `dbcli create-database` command will create a new Oracle Database home.



Note

Bare metal DB systems allow only one database per database home.

- When `--dbhomeid` is provided, the `dbcli create-database` command creates the database using the Oracle home specified. Use the `dbcli list-dbhomes` command to get the dbhomeid. The database home you specify must be empty.
- Oracle Database 12.1 or later databases are supported on both Oracle Automatic Storage Management (ASM) and Oracle ASM Cluster file system (ACFS). The default is Oracle ACFS.
- Oracle Database 11.2 is supported on Oracle ACFS.
- Each database is configured with its own Oracle ACFS file system for the datafiles and uses the following naming convention: `/u02/app/db user/oradata/db` name. The default size of this mount point is 100G.
- Online logs are stored in the `/u03/app/db user/redo/` directory.
- The Oracle Fast Recovery Area (FRA) is located in the `/u03/app/db user/fast_recovery_area` directory.

EXAMPLES

To create a database and be prompted for the password interactively:

```
[root@dbsys ~]# dbcli create-database -n hrdb -c -m -cl OLTP -s odb2 -p pdb1

Password for SYS,SYSTEM and PDB Admin:
{
   "jobId" : "f12485f2-dcbe-4ddf-aee1-de24d37037b6",
   "status" : "Created",
   "message" : null,
   "reports" : [ ],
   "createTimestamp" : "August 08, 2016 03:54:03 AM EDT",
   "description" : "Database service creation with db name: hrdb",
   "updatedTime" : "August 08, 2016 03:54:03 AM EDT"
}
```

To create a database non-interactively, providing the password on the command line:

```
[root@dbsys ~]# dbcli create-database -n crmdb -hm <password> -cl OLTP -s odb2
{
   "jobId" : "30b5e2a6-493b-4461-98b8-78e9a15f8cdd",
   "status" : "Created",
   "message" : null,
   "reports" : [ ],
   "createTimestamp" : "August 08, 2016 03:59:22 AM EDT",
   "description" : "Database service creation with db name: crmdb",
   "updatedTime" : "August 08, 2016 03:59:22 AM EDT"
}
```

DBCLI DELETE-DATABASE

Use the `dbcli delete-database` command to delete a database.



Note

The `dbcli create-database` command is available on bare metal DB systems only.

SYNTAX

```
dbcli  delete-database -i <db_id> -in <db_name> [-fd] [-j] [-h]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-fd	--force	(Optional) Forces the delete operation.
-i	--dbid	The ID of the database to delete. Use the `dbcli list-databases` command to get the database ID.

Parameter	Full Name	Description
-in	--dbName	The name of the database to delete. Use the `dbcli list-databases` command to get the database name.
-j	--json	(Optional) Displays JSON output.

EXAMPLE

The following command deletes the database named 625d9b8a-baea-4994-94e7-4c4a857a17f9:

```
[root@dbsys ~]# dbcli delete-database -i 625d9b8a-baea-4994-94e7-4c4a857a17f9
```

DBCLI DESCRIBE-DATABASE

Use the `dbcli describe-database` command to display database details.

SYNTAX

```
dbcli describe-database -i <db_id> -in <db_name> [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-i	--dbid	The ID of the database to display. Use the `dbcli list-databases` command to get the database ID.
-in	--dbName	The name of the database to display. Use the `dbcli list-databases` command to get the database name.
-j	--json	(Optional) Displays JSON output.

EXAMPLE

The following command displays information for a database named b727bf80-c99e-4846-ac1f-28a81a725df6:

```
[root@dbsys ~]# dbcli describe-dbhome -i b727bf80-c99e-4846-ac1f-28a81a725df6

DB Home details
------------------------------------------------------------
                     ID: b727bf80-c99e-4846-ac1f-28a81a725df6
                   Name: OraDB12102_home1
                Version: 12.1.0.2
          Home Location: /u01/app/orauser/product/12.1.0.2/dbhome_1
                Created: Jun 2, 2016 10:19:23 AM
```

DBCLI LIST-DATABASES

Use the `dbcli list-databases` command to list all databases on the DB system.

SYNTAX

```
dbcli  list-databases [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.

EXAMPLE

The following command displays a list of databases:

```
[root@dbsys ~]# dbcli  list-databases

ID                                       DB Name    DB Version           CDB        Class    Shape
Storage    Status
---------------------------------------- ---------- -------------------- ---------- -------- -------- --
-------- ----------
80ad855a-5145-4f8f-a08f-406c5e4684ff     dbst       12.1.0.2             true       OLTP     odb2
ACFS       Configured
6f4e36ae-120b-4436-b0bf-d0c4aef9f7c9     db11tsta   11.2.0.4             false      OLTP     odb1
ACFS       Configured
d8e31790-84e6-479c-beb0-ef97207091a2     db11tstb   11.2.0.4             false      OLTP     odb1
ACFS       Configured
```

```
cce096c7-737b-447a-baa1-f4c2a330c030     pdbtst     12.1.0.2            true       OLTP     odb1
ACFS       Configured
```

The following command displays the JSON output for a database:

```
[root@dbsys ~]# dbcli list-databases -j
[ {
  "id" : "80ad855a-5145-4f8f-a08f-406c5e4684ff",
  "name" : "dbtst",
  "dbName" : "dbtst",
  "databaseUniqueName" : "dbtst_phx1cs",
  "dbVersion" : "12.1.0.2",
  "dbHomeId" : "2efe7af7-0b70-4e9b-ba8b-71f11c6fe287",
  "instanceOnly" : false,
  "registerOnly" : false,
  "dbId" : "167525515",
  "isCdb" : true,
  "pdBName" : "pdb1",
  "pdbAdminUserName" : "pdbuser",
  "enableTDE" : true,
  "dbType" : "SI",
  "dbTargetNodeNumber" : "0",
  "dbClass" : "OLTP",
  "dbShape" : "odb2",
  "dbStorage" : "ACFS",
  "dbCharacterSet" : {
    "characterSet" : "US7ASCII",
    "nlsCharacterset" : "AL16UTF16",
    "dbTerritory" : "AMERICA",
    "dbLanguage" : "AMERICAN"
  },
  "dbConsoleEnable" : false,
  "backupConfigId" : null,
  "backupDestination" : "NONE",
  "cloudStorageContainer" : null,
  "state" : {
    "status" : "CONFIGURED"
  },
  "createTime" : "November 09, 2016 17:23:05 PM UTC",
  "updatedTime" : "November 09, 2016 18:00:47 PM UTC"
}
```

DBCLI MODIFY-DATABASE

Use the `dbcli modify-database` command to modify a database.

SYNTAX

```
dbcli modify-database -i <db_id> -dh <destination_db_home_id> [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-dh	--destdbhomeid	Destination database home ID.
-h	--help	(Optional) Displays help for using the command.
-i	--databaseid	Database ID.
-j	--json	(Optional) Displays JSON output.

DBCLI RECOVER-DATABASE

Use the `dbcli recover-database` command to recover a database.

SYNTAX

```
dbcli recover-database [-br <json>] [-in <db_name>] [-i <db_id>] [-r <time>] [-t {Latest|PITR|SCN}] [-s]
[-l <location>] [-tp <tde_password>] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-br	--backupReport	(Optional) JSON input for backup report.
-h	--help	(Optional) Displays help for using the command.
-i	--dbid	(Optional) Database resource ID.
-in	--dbName	(Optional) Database name.

Parameter	Full Name	Description
-j	--json	(Optional) Displays JSON output.
-l	--tdeWalletLocation	(Optional) TDE wallet backup location. TDE wallet should be backed up in tar.gz format.
-r	--recoveryTimeStamp	(Required when recovery type is PITR) Recovery timestamp in the format mm/dd/yyyy hh:mi:ss. Default: [ ]
-s	--scn	(Required when recovery type is SCN) SCN.
-t	--recoverytype	(Required when backup report is provided) Recovery type. Possible values are Latest, PITR, and SCN.
-tp	--tdeWalletPassword	(Optional) TDE wallet password.

DBCLI REGISTER-DATABASE

Use the `dbcli register-database` command to register a database that has been migrated to Oracle Cloud Infrastructure. The command registers the database to the dcs-agent so it can be managed by the dcs-agent stack.



Note

The `dbcli register-database` command is not available on 2-node RAC DB systems.

SYNTAX

```
dbcli  register-database -bi <bkup_config_id> -c {OLTP|DSS|IMDB} [-co|-no-co] -s {odb1|odb2|...} -t SI
[-o <db_host_name>] [-tp <password>] -sn <service_name> -p [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-bi	--backupconfigid	Defines the backup configuration ID. Use the `dbcli list-backupconfigs` command to get the ID.
-c	--dbclass	Defines the database class. The options are OLTP, DSS, or IMDB. The default is OLTP. For Enterprise Editions, all three classes are supported. For Standard Edition, only OLTP is supported.
-co -no-co	--dbconsole --no-dbconsole	(Optional) Indicates whether the Database Console is enabled or not. If omitted, the console is not enabled.
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.
-o	--hostname	(Optional) Defines the database host name. The default is `Local host name`.
-p	--syspassword	Defines a strong password for SYS. Specify `-p` with no password. You will be prompted for the password. If you must provide the password in the command, for example in a script, use `-hp <password>` instead of `-p`.
-s	--dbshape	Defines the database sizing template to use for the database. For example, odb1, odb2, and odb3. For more information, see Database Sizing Templates.
-sn	--servicename	Defines the database service name used to build the EZCONNECT string for connecting to the database. The connect string format is `hostname:port/servicename`.

Parameter	Full Name	Description
-t	--dbtype	(Optional) Defines the Database Type as single node (SI). The default value is SI.
-tp	--tdeWalletPassword	(Optional) Password for TDE wallet. Required if TDE is enabled on the migrated database.

EXAMPLE

The following command registers the database with the specified database class, service name, and database sizing template.

```
[root@dbsys ~]# dbcli register-database -c OLTP -s odb1 -sn crmdb.example.com -p
Password for SYS:
{
  "jobId" : "317b430f-ad5f-42ae-bb07-13f053d266e2",
  "status" : "Created",
  "message" : null,
  "reports" : [ ],
  "createTimestamp" : "August 08, 2016 05:55:49 AM EDT",
  "description" : "Database service registration with db service name: crmdb.example.com",
  "updatedTime" : "August 08, 2016 05:55:49 AM EDT"
}
```

DBCLI UPDATE-DATABASE

Use the `dbcli update-database` command to associate a backup configuration with a database.

SYNTAX

```
dbcli update-database -i <db_id> -bi <bkup_config_id> -bin <bkup_config_name> [-id <id>] -in <name> [-
no-ab] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-bi	--backupconfigid	Defines the backup configuration ID. Use the `dbcli list-backupconfigs` command to get the ID.
-bin	--backupconfigname	Defines the backup configuration name for future use. Use the `dbcli list-backupconfigs` command to get the name.
-id	--databaseid	(Optional.) Specifies the DBID, which is a unique 32-bit identification number computed when the database is created. RMAN displays the DBID upon connection to the target database. You can obtain the DBID by querying the V$DATABASE view or the RC_DATABASE and RC_DATABASE_INCARNATION recovery catalog views.
-in	--dbName	Defines the database name to be updated. Use the `dbcli list-databases` command to get the database name.
-h	--help	(Optional) Displays help for using the command.
-i	--dbid	Defines the database ID to be updated. Use the `dbcli list-databases` command to get the database ID.

Parameter	Full Name	Description
-j	--json	(Optional) Displays JSON output.
-no-ab	--noautobackup	(Optional) Disables automatic backups for the specified database.  **Note** Once disabled, automatic backup cannot be re-enabled using the CLI. To re-enable automatic backup, use the Console.

EXAMPLE

The following command associates a backup configuration file with a database:

```
[root@dbsys ~]# dbcli update-database -bi 78a2a5f0-72b1-448f-bd86-cf41b30b64ee -i 71ec8335-113a-46e3-
b81f-235f4d1b6fde
{
  "jobId" : "2b104028-a0a4-4855-b32a-b97a37f5f9c5",
  "status" : "Created",
  "message" : null,
  "reports" : [ ],
  "createTimestamp" : 1467775842977,
  "description" : "update database id:71ec8335-113a-46e3-b81f-235f4d1b6fde",
  "updatedTime" : 1467775842978
}
```

Dbhome Commands

The following commands are available to manage database homes:

- dbcli create-dbhome
- dbcli describe-dbhome

- dbcli delete-dbhome
- dbcli list-dbhomes
- dbcli update-dbhome

DBCLI CREATE-DBHOME

Use the `dbcli create-dbhome` command to create an Oracle Database Home.

SYNTAX

```
dbcli create-dbhome -v <version> [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.
-v	--version	Defines the Database Home version. Specify one of the supported versions: • 18.1.0.0 • 12.2.0.1 • 12.1.0.2 • 11.2.0.4

EXAMPLE

The following command creates an Oracle Database Home version 12.1.0.2:

```
[root@dbsys ~]# dbcli  create-dbhome -v 12.1.0.2
```

DBCLI DESCRIBE-DBHOME

Use the `dbcli describe-dbhome` command to display Oracle Database Home details.

SYNTAX

```
dbcli describe-dbhome -i <db_home_id> [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-i	--dbhomeid	Identifies the database home ID. Use the `dbcli list-dbhomes` command to get the ID.
-j	--json	(Optional) Displays JSON output.

EXAMPLE

The following output is an example of using the display Oracle Database Home details command.

```
[root@dbsys ~]# dbcli describe-dbhome -i 52850389-228d-4397-bbe6-102fda65922b

DB Home details
----------------------------------------------------------------
                     ID: 52850389-228d-4397-bbe6-102fda65922b
                   Name: OraDB12102_home1
                Version: 12.1.0.2
          Home Location: /u01/app/oracle/product/12.1.0.2/dbhome_1
                Created: June 29, 2016 4:36:31 AM UTC
```

DBCLI DELETE-DBHOME

Use the `dbcli delete-dbhome` command to delete a database home from the DB system.

SYNTAX

```
dbcli delete-dbhome -i <db_home_id> [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-i	--dbhomeid	Identifies the database home ID to be deleted. Use the `dbcli list-dbhomes` command to get the ID.
-j	--json	(Optional) Displays JSON output.

DBCLI LIST-DBHOMES

Use the `dbcli list-dbhomes` command to display a list of Oracle Home directories.

SYNTAX

```
dbcli list-dbhomes [-h] [-j]
```

PARAMETER

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.

EXAMPLE

The following command displays a list of Oracle Home directories.

```
[root@dbsys ~]# dbcli  list-dbhomes
ID                                   Name                DB Version Home Location
------------------------------------ ------------------  ---------- ------------------------------------
-----
b727bf80-c99e-4846-ac1f-28a81a725df6 OraDB12102_home1   12.1.0.2
/u01/app/orauser/product/12.1.0.2/dbhome_1
```

DBCLI UPDATE-DBHOME



Tip

Your DB system might not include this newer command. If you have trouble running the command, use the cliadm update-dbcli command to update the database CLI and then retry the command.

Use the `dbcli update-dbhome` command to apply the DBBP bundle patch to a database home. For more information about applying patches, see Patching a DB System.

SYNTAX

```
dbcli update-dbhome -i <db_home_id> -n <node> [--local] [--precheck] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-i	--dbhomeid	The ID of the database home. Use the `dbcli list-dbhomes` command to get the ID.
-j	--json	(Optional) Displays JSON output.
-n	--node	(Optional) Node number to be updated. Use the `dbcli list-nodes` command to get the node number.
	--local	(Optional) Performs the operation on the local node of a multi-node high availability (HA) system. This parameter is not needed to perform the operation on a single-node system.
	--precheck	(Optional) Runs precheck operations to check prerequisites.

EXAMPLE

The following commands update the database home and show the output from the update job:

```
[root@dbsys ~]# dbcli update-dbhome -i e1877dac-a69a-40a1-b65a-d5e190e671e6
{
  "jobId" : "493e703b-46ef-4a3f-909d-bbd123469bea",
  "status" : "Created",
  "message" : null,
  "reports" : [ ],
  "createTimestamp" : "January 19, 2017 10:03:21 AM PST",
  "resourceList" : [ ],
  "description" : "DB Home Patching: Home Id is e1877dac-a69a-40a1-b65a-d5e190e671e6",
  "updatedTime" : "January 19, 2017 10:03:21 AM PST"
}

# dbcli describe-job -i 493e703b-46ef-4a3f-909d-bbd123469bea

Job details
----------------------------------------------------------------
                     ID:  493e703b-46ef-4a3f-909d-bbd123469bea
            Description:  DB Home Patching: Home Id is e1877dac-a69a-40a1-b65a-d5e190e671e6
                 Status:  Running
                Created:  January 19, 2017 10:03:21 AM PST
                Message:

Task Name                                Start Time                          End Time
        Status
---------------------------------------- ----------------------------------- ---------------------------
-------- ----------
Create Patching Repository Directories   January 19, 2017 10:03:21 AM PST    January 19, 2017 10:03:21
AM PST    Success
Download latest patch metadata           January 19, 2017 10:03:21 AM PST    January 19, 2017 10:03:21
AM PST    Success
Update System version                    January 19, 2017 10:03:21 AM PST    January 19, 2017 10:03:21
AM PST    Success
Update Patching Repository               January 19, 2017 10:03:21 AM PST    January 19, 2017 10:03:31
AM PST    Success
Opatch updation                          January 19, 2017 10:03:31 AM PST    January 19, 2017 10:03:31
AM PST    Success
Patch conflict check                     January 19, 2017 10:03:31 AM PST    January 19, 2017 10:03:31
AM PST    Running
```

Dbstorage Commands

The following commands are available to manage database storage:

- dbcli list-dbstorages
- dbcli describe-dbstorage
- dbcli create-dbstorage
- dbcli delete-dbstorage

DBCLI LIST-DBSTORAGES

Use the `dbcli list-dbstorages` command to list the database storage in the DB system.

SYNTAX

```
dbcli list-dbstorages [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.

EXAMPLE

The following command displays details about database storage:

```
[root@dbsys ~]# dbcli list-dbstorages

ID                                       Type   DBUnique Name        Status
---------------------------------------- ------ -------------------- ----------
afb4a1ce-d54d-4993-a149-0f28c9fb33a4     Acfs   db1_2e56b3a9b815     Configured
d81e8013-4551-4d10-880b-d1a796bca1bc     Acfs   db11xp               Configured
```

DBCLI DESCRIBE-DBSTORAGE

Use the `dbcli describe-dbstorage` command to show detailed information about a specific database storage resource.

SYNTAX

```
dbcli describe-dbstorage -i <db_storage_id> [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-i	--id	Defines the database storage ID. Use the `dbcli list-dbstorages` command to get the database storage ID.
-j	--json	(Optional) Displays JSON output.

EXAMPLE

The following command displays the database storage details for 105a2db2-625a-45ba-8bdd-ee46da0fd83a:

```
[root@dbsys ~]# dbcli describe-dbstorage -i 105a2db2-625a-45ba-8bdd-ee46da0fd83a

DBStorage details
----------------------------------------------------------------

                     ID: 105a2db2-625a-45ba-8bdd-ee46da0fd83a
                DB Name: db1
          DBUnique Name: db1
         DB Resource ID: 439e7bd7-f717-447a-8046-08b5f6493df0
           Storage Type:
          DATA Location: /u02/app/oracle/oradata/db1
          RECO Location: /u03/app/oracle/fast_recovery_area/
          REDO Location: /u03/app/oracle/redo/
                  State: ResourceState(status=Configured)
                Created: July 3, 2016 4:19:21 AM UTC
            UpdatedTime: July 3, 2016 4:41:29 AM UTC
```

DBCLI CREATE-DBSTORAGE

Use the `dbcli create-dbstorage` command to create the database storage layout without creating the complete database. This is useful for database migration and standby database

creation.

SYNTAX

```
dbcli create-dbstorage -n <db_name> [-u <db_unique_name>] [-r {ACFS|ASM}] [-s <datasize>] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.
-n	--dbname	Defines the database name. The database name must begin with an alphabetic character and can contain a maximum of eight alphanumeric characters. Special characters are not permitted.
-r	--dbstorage	(Optional) Defines the type of database storage as ACFS or ASM. The default value is ASM.
-s	--dataSize	(Optional) Defines the data size in GBs. The minimum size is 10GB. The default size is 100GB.
-u	--databaseUniqueName	(Optional) Defines the unique name for the database. The default is the database name specified in `--dbname`.

EXAMPLE

The following command creates database storage with a storage type of ACFS:

```
[root@dbsys ~]# dbcli create-dbstorage -r ACFS -n testdb -u testdbname

{
  "jobId" : "5884a77a-0577-414f-8c36-1e9d8a1e9cee",
  "status" : "Created",
  "message" : null,
  "reports" : [ ],
```

```
  "createTimestamp" : 1467952215102,
  "description" : "Database storage service creation with db name: testdb",
  "updatedTime" : 1467952215103
}
```

DBCLI DELETE-DBSTORAGE

Use the `dbcli delete-dbstorage` command to delete database storage that is not being used by the database. A error occurs if the resource is in use.

SYNTAX

```
dbcli delete-dbstorage -i <dbstorageID> [-h] [-j]
```

PARAMETERS

Parameter	Parameter	Description
-h	--help	(Optional) Displays help for using the command.
-i	--id	The database storage ID to delete. Use the `dbcli list-dbstorages` command to get the database storage ID.
-j	--json	(Optional) Displays JSON output.

EXAMPLE

The following command deletes the specified database storage:

```
[root@dbsys ~]# dbcli delete-dbstorage -i f444dd87-86c9-4969-a72c-fb2026e7384b

{
  "jobId" : "467c9388-18c6-4e1a-8655-2fd3603856ef",
  "status" : "Running",
  "message" : null,
  "reports" : [ ],
  "createTimestamp" : 1467952336843,
  "description" : "Database storage service deletion with id: f444dd87-86c9-4969-a72c-fb2026e7384b",
  "updatedTime" : 1467952336856
}
```

Dgconfig Commands

DBCLI LIST-DGCONFIGS

Use the `dbcli list-dgconfigs` command to list DG configurations.

SYNTAX

```
dbcli list-dgconfigs [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.

Featuretracking Commands

DBCLI LIST-FEATURETRACKING

Use the `dbcli list-featuretracking` command to list tracked features.

SYNTAX

```
dbcli list-featuretracking[-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.

Job Commands

The following commands are available to manage jobs:

- dbcli describe-job
- dbcli list-jobs

DBCLI DESCRIBE-JOB

Use the `dbcli describe-job` command to display details about a specific job.

SYNTAX

```
dbcli describe-job -i <job_id> [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-i	--jobid	Identifies the job. Use the `dbcli list-jobs` command to get the jobid.
-j	--json	(Optional) Displays JSON output.

EXAMPLE

The following command displays details about the specified job ID:

```
[root@dbsys ~]# dbcli describe-job -i 74731897-fb6b-4379-9a37-246912025c17

Job details
----------------------------------------------------------------
                     ID:  74731897-fb6b-4379-9a37-246912025c17
            Description:  Backup service creation with db name: dbtst
                 Status:  Success
                Created:  November 18, 2016 8:33:04 PM UTC
                Message:

Task Name                                Start Time                          End Time
        Status
---------------------------------------- ----------------------------------- ---------------------------
-------- ----------
Backup Validations                       November 18, 2016 8:33:04 PM UTC    November 18, 2016 8:33:13
```

```
PM UTC    Success
validate recovery window                November 18, 2016 8:33:13 PM UTC    November 18, 2016 8:33:17
PM UTC    Success
Db cross check                          November 18, 2016 8:33:17 PM UTC    November 18, 2016 8:33:23
PM UTC    Success
Database Backup                         November 18, 2016 8:33:23 PM UTC    November 18, 2016 8:34:22
PM UTC    Success
Backup metadata                         November 18, 2016 8:34:22 PM UTC    November 18, 2016 8:34:22
PM UTC    Success
```

DBCLI LIST-JOBS

Use the `dbcli list-jobs` command to display a list of jobs, including the job IDs, status, and the job

created date and time stamp.

SYNTAX

```
dbcli list-jobs [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.

EXAMPLE

The following command displays a list of jobs:

```
[root@dbsys ~]# dbcli list-jobs

ID                                       Description
            Created                             Status
---------------------------------------- --------------------------------------------------------------------
------------ ---------------------------------- ----------
0a362dac-0339-41b5-9c9c-4d229e363eaa     Database service creation with db name: db11
            November 10, 2016 11:37:54 AM UTC   Success
9157cc78-b487-4ee9-9f46-0159f10236e4     Database service creation with db name: jhfpdb
```

```
            November 17, 2016 7:19:59 PM UTC    Success
013c408d-37ca-4f58-a053-02d4efdc42d0     create backup config:myBackupConfig
            November 18, 2016 8:28:14 PM UTC    Success
921a54e3-c359-4aea-9efc-6ae7346cb0c2     update database id:80ad855a-5145-4f8f-a08f-406c5e4684ff
            November 18, 2016 8:32:16 PM UTC    Success
74731897-fb6b-4379-9a37-246912025c17     Backup service creation with db name: dbtst
            November 18, 2016 8:33:04 PM UTC    Success
40a227b1-8c47-46b9-a116-48cc1476fc12     Creating a report for database 80ad855a-5145-4f8f-a08f-
406c5e4684ff         November 18, 2016 8:41:39 PM UTC    Success
```

Latestpatch Command

DBCLI DESCRIBE-LATESTPATCH



Tip

Your DB system might not include this newer command. If you have trouble running the command, use the cliadm update-dbcli command to update the database CLI and then retry the command.



Note

The `dbcli describe-latestpatch` command is not available on 2-node RAC DB systems. Patching 2-node systems from Object Storage is not supported.

Use the `dbcli describe-latestpatch` command show the latest patches applicable to the DB system and available in Oracle Cloud Infrastructure Object Storage.

This command requires a valid Object Storage credentials configuration. Use the dbcli create-bmccredential command to create the configuration if you haven't already done so. If the configuration is missing or invalid, the command fails with the error: `Failed to connect to the object store. Please provide valid details.`

For more information about updating the CLI, creating the credentials configuration, and applying patches, see Patching a DB System.

SYNTAX

```
dbcli describe-latestpatch [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.

EXAMPLE

The following command displays patches available in the object store:

```
[root@dbsys ~]# dbcli describe-latestpatch

componentType   availableVersion
--------------- --------------------
gi              12.1.0.2.161018
db              11.2.0.4.161018
db              12.1.0.2.161018
oak             12.1.2.10.0
```

Logcleanjob Commands

The following commands are available to manage log cleaning jobs:

- dbcli create-logCleanJob
- dbcli describe-logCleanJob
- dbcli list-logCleanJobs

DBCLI CREATE-LOGCLEANJOB

Use the `dbcli create-logCleanJob` command to create a log cleaning job.

SYNTAX

```
dbcli create-logCleanJob [-c {gi|database|dcs}] [-o <number>] [u {Day|Hour|Minute}] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-c	--components	(Optional) Components. Possible values are gi, database, and dcs. Separate multiple values by commas.
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.
-o	--olderThan	(Optional) Quantity portion of time interval. Default: 30. Cleans logs older than the specified time interval (`-o` and `-u`).
-u	--unit	(Optional) Unit portion of time interval. Possible values: Day, Hour, or Minute. Default: Day. Cleans logs older than the specified time interval (`-o` and `-u`).

DBCLI DESCRIBE-LOGCLEANJOB

Use the `dbcli describe-logCleanJob` command to display the summary for a log cleaning job.

SYNTAX

```
dbcli describe-logCleanJob -i <job_id> [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-i	--jobid	ID of log cleaning job for which to display the summary.
-j	--json	(Optional) Displays JSON output.

DBCLI LIST-LOGCLEANJOBS

Use the `dbcli list-logCleanJobs` command to list log cleaning jobs.

SYNTAX

```
dbcli list-logCleanJobs [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.

Logspaceusage Command

DBCLI LIST-LOGSPACEUSAGE

Use the `dbcli list-logSpaceUsage` command to list log space usage.

SYNTAX

```
dbcli list-logSpaceUsage [-c {gi|database|dcs}] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-c	--components	(Optional) Components. Possible values: gi, database, and dcs. Separate multiple values by commas.
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.

Netsecurity Commands

The following commands are available to manage network encryption on the DB system:

- dbcli describe-netsecurity
- dbcli update-netsecurity

DBCLI DESCRIBE-NETSECURITY

Use the `dbcli describe-netsecurity` command to display the current network encryption setting for a database home.

SYNTAX

```
dbcli describe-netsecurity -H <db_home_id> [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-H	--dbHomeId	Defines the database home ID. Use the `dbcli list-dbhomes` command to get the dbhomeid.
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.

EXAMPLE

The following command displays the encryption setting for specified database home:

```
[root@dbsys ~]# dbcli describe-netsecurity -H 16c96a9c-f579-4a4c-a645-8d4d22d6889d

NetSecurity Rules
-----------------------------------------------------------------
         DatabaseHomeID: 16c96a9c-f579-4a4c-a645-8d4d22d6889d

                   Role: Server
   EncryptionAlgorithms: AES256 AES192 AES128
    IntegrityAlgorithms: SHA1
         ConnectionType: Required

                   Role: Client
   EncryptionAlgorithms: AES256 AES192 AES128
```

```
    IntegrityAlgorithms: SHA1
         ConnectionType: Required
```

DBCLI UPDATE-NETSECURITY

Use the `dbcli update-netsecurity` command to update the Oracle Net security configuration on the DB system.

SYNTAX

```
dbcli update-netsecurity {-c|-s} -t {REJECTED|ACCEPTED|REQUESTED|REQUIRED} -H db_home_id> -e
{AES256|AES192|AES128} -i {SHA1|SHA512|SHA384|SHA256} [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-c	--client	Indicates that the specified data encryption or data integrity configuration is for the client. (`--client` and `--server` are mutually exclusive.)
-e	--encryptionAlgorithms	Defines the algorithm to be used for encryption. Specify either AES256, AES192, or AES128.
-H	--dbHomeId	Defines the database home ID. Use the `dbcli list-dbhomes` command to get the dbHomeId.
-h	--help	(Optional) Displays help for using the command.
-i	--integrityAlgorithms	Defines the algorithm to be used for integrity. Specify either SHA1, SHA512, SHA384, or SHA256. For Oracle Database 11g, the only accepted value is SHA1.
-j	--json	(Optional) Displays JSON output.

Parameter	Full Name	Description
-s	--server	Indicates that the specified data encryption or data integrity configuration is for the server. (`--client` and `--server` are mutually exclusive.)
-t	--connectionType	Specifies how Oracle Net Services data encryption or data integrity is negotiated with clients. The following values are listed in the order of increasing security: REJECTED - Do not enable data encryption or data integrity, even if required by the client. ACCEPTED - Enable data encryption or data integrity if required or requested by the client. REQUESTED - Enable data encryption or data integrity if the client permits it. REQUIRED - Enable data encryption or data integrity or preclude the connection. For detailed information about network data encryption and integrity, see https://docs.oracle.com/database/121/DBSEG/asoconfg.htm#DBSEG1047.

EXAMPLE

The following command updates the connection type to ACCEPTED:

```
[root@dbsys ~]# dbcli update-netsecurity -H a2ffbb07-c9c0-4467-a458-bce4d3b76cd5 -t ACCEPTED
```

Node Command

DBCLI LIST-NODES

Use the `dbcli list-nodes` command to display a list of nodes, including the node numbers.

SYNTAX

```
dbcli list-nodes [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.

EXAMPLE

The following command displays a list of nodes:

```
[root@dbsys ~]# dbcli list-nodes
node Number node Name                                ilom Name            IP Address          Subnet              Mask Gate
---------- ---------------------------------------- -------------------- ------------------- ------------
------ ------------------
0          rac21                                     N/A                  N/A                 N/A                 N/A
1          rac22                                     N/A                  N/A                 N/A                 N/A
```

Objectstoreswift Commands

You can back up a database to an existing bucket in the Oracle Cloud Infrastructure Object Storage service by using the dbcli create-backup command, but first you'll need to:

1. Create an object store on the DB system, which contains the endpoint and credentials to access Object Storage, by using the dbcli create-objectstoreswift command.
2. Create a backup configuration that refers to the object store ID and the bucket name by using the dbcli create-backupconfig command.
3. Associate the backup configuration with the database by using the dbcli update-database command.

The following commands are available to manage object stores.

- dbcli create-objectstoreswift
- dbcli describe-objectstoreswift
- dbcli list-objectstoreswifts

DBCLI CREATE-OBJECTSTORESWIFT

Use the `dbcli create-objectstoreswift` command to create an object store.

SYNTAX

```
dbcli create-objectstoreswift -n <object_store_name> -t <tenant_name> -u <user_name> -e
https://swiftobjectstorage.<region_name>.oraclecloud.com/v1 -p [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-e	--endpointurl	The following endpoint URL. `https://swiftobjectstorage.<region_name>.oraclecloud.com/v1` See Regions and Availability Domains for region name strings.
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.
-n	--name	The name for the object store to be created.

Parameter	Full Name	Description
-p	--swiftpassword	The auth token that you generated by using the Console or IAM API. For information about generating an auth token for use with Swift, see Managing User Credentials. This is **not** the password for the Oracle Cloud Infrastructure user. Specify `-p` (with no password) to be prompted. Specify `-hp "<password>"` in quotes to provide the password (auth token) in the command.
-t	--tenantname	The case-sensitive tenant name that you specify when signing in to the Console.
-u	--username	The user name for the Oracle Cloud Infrastructure user account, for example: `-u djones@example.com` This is the user name you use to sign in to the Console. The user name must have tenancy-level access to the Object Storage. An easy way to do this is to add the user name to the Administrators group. However, that allows access to *all* of the cloud services. Instead, an administrator can create a policy that allows tenancy-level access to just Object Storage. The following is an example of such a policy. `Allow group DBAdmins to manage buckets in tenancy` `Allow group DBAdmins to manage objects in tenancy` For more information about adding a user to a group, see Managing Groups. For more information about policies, see Getting Started with Policies.

EXAMPLE

The following command creates an object store and prompts for the Swift password:

```
[root@dbsys ~]# dbcli create-objectstoreswift -n r2swift -t CompanyABC -u djones@example.com -e
https://swiftobjectstorage.<region_name>.oraclecloud.com/v1 -p
Password for Swift:
{
  "jobId" : "c565bb71-f67b-4fab-9d6f-a34eae36feb7",
  "status" : "Created",
  "message" : "Create object store swift",
  "reports" : [ ],
  "createTimestamp" : "January 19, 2017 11:11:33 AM PST",
  "resourceList" : [ {
    "resourceId" : "8a0fe039-f5d4-426a-8707-256c612b3a30",
    "resourceType" : "ObjectStoreSwift",
    "jobId" : "c565bb71-f67b-4fab-9d6f-a34eae36feb7",
    "updatedTime" : "January 19, 2017 11:11:33 AM PST"
  } ],
  "description" : "create object store:biyanr2swift",
  "updatedTime" : "January 19, 2017 11:11:33 AM PST"
}
```

DBCLI DESCRIBE-OBJECTSTORESWIFT

Use the `dbcli describe-objectstoreswift` command to display details about an object store.

SYNTAX

```
dbcli describe-objectstoreswift -i <object_store_swift_id> -in <object_store_swift_name> [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-i	--objectstoreswiftid	The object store ID. Use the `dbcli list-objectstoreswifts` command to get the ID.

Parameter	Full Name	Description
-in	--objectstoreswiftName	The object store name. Use the `dbcli list-objectstoreswifts` command to get the name.
-j	--json	(Optional) Displays JSON output.

EXAMPLE

The following command displays details about an object store:

```
[root@dbsys ~]# dbcli describe-objectstoreswift -i 910e9e2d-25b4-49b4-b88e-ff0332f7df87
Object Store details
----------------------------------------------------------------
                     ID: 910e9e2d-25b4-49b4-b88e-ff0332f7df87
                   Name: objstrswift15
               UserName: djones@example.com
             TenantName: CompanyABC
           endpoint URL: https://swiftobjectstorage.<region_name>.oraclecloud.com/v1
            CreatedTime: November 16, 2016 11:25:34 PM UTC
            UpdatedTime: November 16, 2016 11:25:34 PM UTC
```

DBCLI LIST-OBJECTSTORESWIFTS

Use the `dbcli list-objectstoreswifts` command to list the object stores on a DB system.

SYNTAX

```
dbcli list-objectstoreswifts [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.

EXAMPLE

The following command lists the object stores on the DB system:

```
[root@dbsys ~]# dbcli list-objectstoreswifts

ID                                       Name                 UserName              TenantName     Url
                                                          createTime
---------------------------------------- -------------------- --------------------  -------------- -----
- -------------------------------------------------- ----------------------------------
2915bc6a-6866-436a-a38c-32302c7c4d8b     swiftobjstr1         djones@example.com    LargeComputers
https://swiftobjectstorage.<region_name>.oraclecloud.com/v1 November 10, 2016 8:42:18 PM UTC
910e9e2d-25b4-49b4-b88e-ff0332f7df87     objstrswift15        djones@example.com    LargeComputers
https://swiftobjectstorage.<region_name>.oraclecloud.com/v1 November 16, 2016 11:25:34 PM UTC
```

Pendingjob Command

DBCLI LIST-PENDINGJOBS

Use the `dbcli list-pendingjobs` command to display a list of pending jobs.

SYNTAX

```
dbcli list-pendingjobs [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.

Rmanbackupreport Commands

The following commands are available to manage RMAN backup reports:

- dbcli create-rmanbackupreport
- dbcli delete-rmanbackupreport
- dbcli describe-rmanbackupreport
- dbcli list-rmanbackupreports

DBCLI CREATE-RMANBACKUPREPORT

Use the `dbcli create-rmanbackupreport` command to create an RMAN backup report.

SYNTAX

```
dbcli create-rmanbackupreport -w {summary|detailed} -rn <name> [-i <db_id>] [-in <db_name>] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-i	--dbid	(Optional) Database resource ID.
-in	--dbname	(Optional) Database resource name.
-j	--json	(Optional) Displays JSON output.
-rn	--rptname	RMAN backup report name. Maximum number of characters: 30. Wrap name in single quotes when special characters are used.
-w	--reporttype	RMAN backup report type. Possible values: summary or detailed.

DBCLI DELETE-RMANBACKUPREPORT

Use the `dbcli delete-rmanbackupreport` command to delete an RMAN backup report.

SYNTAX

```
dbcli delete-rmanbackupreport [-d <db_id>] [-dn <db_name>] [-n <number>] [-i <rpt_id>] [-in <rpt_name>]
[-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-d	--dbid	(Optional) Database resource ID.
-dn	--dbname	(Optional) Database resource name.
-h	--help	(Optional) Displays help for using the command.
-i	--reportid	(Optional) RMAN backup report ID
-in	--rptname	(Optional) RMAN backup report name
-j	--json	(Optional) Displays JSON output.
-n	--numofday	(Optional) Number of days since created (provided with Database ID/Database Name)

DBCLI DESCRIBE-RMANBACKUPREPORT

Use the `dbcli describe-rmanbackupreport` command to

SYNTAX

```
dbcli describe-rmanbackupreport [-i <rpt_id>] [-in <rpt_name>] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-i	--id	(Optional) RMAN backup report ID
-in	--name	(Optional) RMAN backup report name
-j	--json	(Optional) Displays JSON output.

DBCLI LIST-RMANBACKUPREPORTS

Use the `dbcli list-rmanbackupreports` command to

SYNTAX

```
dbcli list-rmanbackupreports [-i <db_id>] [-in <db_name>] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-i	--dbid	(Optional) Database resource ID.
-in	--dbName	(Optional) Database resource name.
-j	--json	(Optional) Displays JSON output.

Schedule Commands

The following commands are available to manage schedules:

- dbcli describe-schedule
- dbcli list-schedules
- dbcli update-schedule

DBCLI DESCRIBE-SCHEDULE

Use the `dbcli describe-schedule` command to describe a schedule.

SYNTAX

```
dbcli describe-schedule -i <id> [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-i	--scheduleid	Schedule ID.
-j	--json	(Optional) Displays JSON output.

DBCLI LIST-SCHEDULES

Use the `dbcli list-schedules` command to list schedules.

SYNTAX

```
dbcli list-schedules [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.

DBCLI UPDATE-SCHEDULE

Use the `dbcli update-schedule` command to update a schedule.

SYNTAX

```
dbcli update-schedule -i <id> [-x <expression>] [-t <description>] [-d] [-e] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-d	--disable	(Optional) Disables the schedule.
-e	--enable	(Optional) Enables the schedule.
-h	--help	(Optional) Displays help for using the command.
-i	--scheduleid	Schedule ID.
-j	--json	(Optional) Displays JSON output.
-t	--description	(Optional) Description
-x	--cronExpression	(Optional) Cron expression. Use cronmaker.com to generate a valid cron expression.

Scheduledexecution Command

DBCLI LIST-SCHEDULEDEXECUTIONS

Use the `dbcli list-scheduledExecutions` command to list scheduled executions.

SYNTAX

```
dbcli list-scheduledExecutions [-e <execution_id>] [-i <schedule_id>] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-e	--executionid	(Optional) Execution ID.
-h	--help	(Optional) Displays help for using the command.

Parameter	Full Name	Description
-i	--scheduleid	(Optional) Schedule ID.
-j	--json	(Optional) Displays JSON output.

Server Command

DBCLI UPDATE-SERVER



Tip

Your DB system might not include this newer command. If you have trouble running the command, use the cliadm update-dbcli command to update the database CLI and then retry the command.

Use the `dbcli update-server` command to apply patches to the server components in the DB system. For more information about applying patches, see Patching a DB System.

SYNTAX

```
dbcli update-server [-n <number>] [--local] [--precheck] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.

Parameter	Full Name	Description
-l	--local	(Optional) Performs the operation on the local node of a multi-node high availability (HA) system. This parameter is not needed to perform the operation on a single-node system.
-n	--node	(Optional) Node number to be updated. Use the `dbcli list-nodes` command to get the node number.
-p	--precheck	(Optional) Runs precheck operations to check prerequisites.

EXAMPLES

The following commands update the server and show the output from the update job:

```
[root@dbsys ~]# dbcli update-server
{
  "jobId" : "9a02d111-e902-4e94-bc6b-9b820ddf6ed8",
  "status" : "Created",
  "reports" : [ ],
  "createTimestamp" : "January 19, 2017 09:37:11 AM PST",
  "resourceList" : [ ],
  "description" : "Server Patching",
  "updatedTime" : "January 19, 2017 09:37:11 AM PST"
}

# dbcli describe-job -i 9a02d111-e902-4e94-bc6b-9b820ddf6ed8

Job details
----------------------------------------------------------------
                     ID:  9a02d111-e902-4e94-bc6b-9b820ddf6ed8
            Description:  Server Patching
                 Status:  Running
                Created:  January 19, 2017 9:37:11 AM PST
                Message:

Task Name                                Start Time                          End Time
        Status
---------------------------------------- ----------------------------------- ---------------------------
-------- ----------
Create Patching Repository Directories   January 19, 2017 9:37:11 AM PST     January 19, 2017 9:37:11 AM
```

```
PST     Success
Download latest patch metadata          January 19, 2017 9:37:11 AM PST     January 19, 2017 9:37:11 AM
PST     Success
Update System version                   January 19, 2017 9:37:11 AM PST     January 19, 2017 9:37:11 AM
PST     Success
Update Patching Repository              January 19, 2017 9:37:11 AM PST     January 19, 2017 9:38:35 AM
PST     Success
oda-hw-mgmt upgrade                     January 19, 2017 9:38:35 AM PST     January 19, 2017 9:38:58 AM
PST     Success
Opatch updation                         January 19, 2017 9:38:58 AM PST     January 19, 2017 9:38:58 AM
PST     Success
Patch conflict check                    January 19, 2017 9:38:58 AM PST     January 19, 2017 9:42:06 AM
PST     Success
apply clusterware patch                 January 19, 2017 9:42:06 AM PST     January 19, 2017 10:02:32
AM PST    Success
Updating GiHome version                 January 19, 2017 10:02:32 AM PST    January 19, 2017 10:02:38
AM PST    Success
```

The following command updates node 0 of the server only, with precheck:

```
# dbcli update-server -n 0 -p
{
  "jobId" : "3e2a1e3c-83d3-4101-86b8-4d525f3f8c18",
  "status" : "Created",
  "message" : null,
  "reports" : [ ],
  "createTimestamp" : "April 26, 2019 06:07:27 AM UTC",
  "resourceList" : [ ],
  "description" : "Server Patching Prechecks",
  "updatedTime" : "April 26, 2019 06:07:27 AM UTC"
}
```

System Command

DBCLI DESCRIBE-SYSTEM

Use the `dbcli describe-system` command to display details about the system. On a 2-node RAC DB system, the command provides information about the local node.

SYNTAX

```
dbcli describe-system [-b] [-d] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-b	--bom	(Optional) Displays BOM information.
-d	--details	(Optional) Displays additional information about the DB system, including dcs CLI and agent version information.
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.

TDE Commands

The following commands are available to manage TDE-related items (backup reports, keys, and wallets):

- dbcli list-tdebackupreports
- dbcli update-tdekey
- dbcli recover-tdewallet

DBCLI LIST-TDEBACKUPREPORTS

Use the `dbcli list-tdebackupreports` command to list backup reports for TDE wallets.

SYNTAX

```
dbcli  list-tdebackupreports [-i <db_id>] [-in <db_name>] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-i	--dbResid	(Optional) Displays the TDE Wallet backup reports for the specified database resource ID. Use the `dbcli list-databases` command to get the database resource ID.
-in	--dbResname	(Optional) Displays the TDE Wallet backup reports for the specified database resource name. Use the `dbcli list-databases` command to get the database resource name.
-j	--json	(Optional) Displays JSON output.

EXAMPLE

The following command lists the backup reports for TDE wallets:

```
[root@dbsys ~]# dbcli list-tdebackupreports
DbResID                                  OraDbId                 BackupLocation
---------------------------------------- ----------------------  ----------------------------------------
538ca5b1-654d-4418-8ce1-f49b6c987a60   1257156075              https://swiftobjectstorage.us-phoenix-
1.oraclecloud.com/v1/dbaasimage/backuptest/host724007/tdewallet/Testdb5/1257156075/2017-08-17/TDEWALLET_
BMC60_2017-08-17_10-58-17.0990.tar.gz
538ca5b1-9fb2-4245-b157-6e25d7c988c5   704287483               https://swiftobjectstorage.us-phoenix-
1.oraclecloud.com/v1/dbaasimage/backuptest/host724007/tdewallet/Testdb1/704287483/2017-08-17/TDEWALLET_
AUTO_2017-08-17_11-03-25.0953.tar.gz
538ca5b1-9fb2-4245-b157-6e25d7c988c5   704287483               https://swiftobjectstorage.us-phoenix-
1.oraclecloud.com/v1/dbaasimage/backuptest/host724007/tdewallet/Testdb1/704287483/2017-08-17/TDEWALLET_
BMC62_2017-08-17_11-04-41.0264.tar.gz
19714ffa-de1b-4433-9188-c0592887e609   1157116855              https://swiftobjectstorage.us-phoenix-
1.oraclecloud.com/v1/dbaasimage/backuptest/host724007/tdewallet/Testdb7/1157116855/2017-08-17/TDEWALLET_
AUTO_2017-08-17_11-57-47.0605.tar.gz
```

DBCLI UPDATE-TDEKEY

Use the `dbcli update-tdekey` command to update the TDE encryption key inside the TDE wallet. You can update the encryption key for Pluggable Databases (if -pdbNames are

specified), and/or the Container Database (if -rootDatabase is specified).

SYNTAX

```
dbcli update-tdekey -i <db_id> -p [-all] -n <pdbname1,pdbname2> [-r|-no-r] -t <tag_name> [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-all	--allPdbNames	(Optional) Flag to rotate (update) all PDB names. To update all instead of specified PDB names, use this parameter instead of `-n`. Default: false.
-i	--databaseId	Defines the database ID for which to update the key.
-p	--password	Defines the TDE Admin wallet password. Specify `-p` with no password. You will be prompted for the password. If you must provide the password in the command, for example in a script, use `-hp <password>` instead of `-p`.
-n	--pdbNames	Defines the PDB names to be rotated (updated).
-r -no-r	--rootDatabase --no-rootDatabase	Indicates whether to rotate the key for the root database if it is a container database.
-t	-tagName	Defines the TagName used to backup the wallet. The default is OdaRotateKey.
-h	--help	(Optional) Displays help for using the command.
-j	--json	(Optional) Displays JSON output.

EXAMPLE

The following command updates the key for pdb1 and pdb2 only:

```
[root@dbsys ~]# dbcli update-tdekey -dbid ee3eaab6-a45b-4e61-a218-c4ba665503d9 -p -n pdb1,pdb2

TDE Admin wallet password:
{
  "jobId" : "08e5edb1-42e1-4d16-a47f-783c0afa4778",
  "status" : "Created",
  "message" : null,
  "reports" : [ ],
  "createTimestamp" : 1467876407035,
  "description" : "TDE update",
  "updatedTime" : 1467876407035
}
```

The following command updates pdb1, pdb2, and the container database:

```
[root@dbsys ~]# dbcli update-tdekey -dbid ee3eaab6-a45b-4e61-a218-c4ba665503d9 -p -n pdb1,pdb2 -r

TDE Admin wallet password:
{
  "jobId" : "c72385f0-cd81-42df-a8e8-3a1e7cab1278",
  "status" : "Created",
  "message" : null,
  "reports" : [ ],
  "createTimestamp" : 1467876433783,
  "description" : "TDE update",
  "updatedTime" : 1467876433783
}
```

DBCLI RECOVER-TDEWALLET

Use the `dbcli recover-tdewallet` command to recover a TDE wallet.

SYNTAX

```
dbcli recover-tdewallet -in <db_name> -tp <password> [-l <location>] [-h] [-j]
```

PARAMETERS

Parameter	Full Name	Description
-h	--help	(Optional) Displays help for using the command.
-in	--dbName	Database name.
-j	--json	(Optional) Displays JSON output.
-l	--tdeWalletBackuplocation	(Optional) TDE wallet backup location. TDE wallet should b ebacked up in tar.gz format.
-tp	--tdeWalletPassword	Defines the TDE Admin wallet password.

Admin Commands

The following commands are to perform administrative actions on the DB system:

- dbadmcli manage diagcollect
- dbadmcli power
- dbadmcli power disk status
- dbadmcli show controller
- dbadmcli show disk
- dbadmcli show diskgroup
- dbadmcli show env_hw (environment type and hardware version)
- dbadmcli show fs (file system details)
- dbadmcli show storage
- dbadmcli stordiag

DBADMCLI MANAGE DIAGCOLLECT

Use the `dbadmcli manage diagcollect` command to collect diagnostic information about a DB system for troubleshooting purposes, and for working with Oracle Support Services.

SYNTAX

```
dbadmcli manage diagcollect --storage [-h]
```

PARAMETERS

Parameter	Description
-h	(Optional) Displays help for using the command.
--storage	Collects all of the logs for any storage issues.

EXAMPLE

```
[root@dbsys ~]# dbadmcli manage diagcollect --storage
Collecting storage log data. It will take a while, please wait...
Collecting oak data. It will take a while, please wait...
tar: Removing leading `/' from member names
tar: /opt/oracle/oak/onecmd/tmp/OakCli-Command-Output.log: file changed as we read it

Logs are collected to : /opt/oracle/oak/log/dbsys/oakdiag/oakStorage-dbsys-20161118_2101.tar.gz
```

DBADMCLI POWER

Use the `dbadmcli power` command to power a disk on or off.



Note

The `dbadmcli power` command is not available on 2-node RAC DB systems.

SYNTAX

```
dbadmcli power {-on|-off} <name> [-h]
```

PARAMETERS

Parameter	Description
-h	(Optional) Displays help for using the command.
name	Defines the disk resource name. The resource name format is pd_[0..3]. Use the `dbadmcli show disk` command to get the disk resource name.
-off	Powers off the disk.
-on	Powers on the disk.

DBADMCLI POWER DISK STATUS

Use the `dbadmcli power disk status` command to display the current power status of a disk.

SYNTAX

```
dbadmcli power disk status <name> [-h]
```

PARAMETERS

Parameter	Description
-h	(Optional) Displays help for using the command.
name	Identifies a specific disk resource name. The resource name format is pd_[0..3]. For example, pd_01.

EXAMPLE

```
[root@dbsys ~]# dbadmcli power disk status pd_00

The disk is powered ON
```

DBADMCLI SHOW CONTROLLER

Use the `dbadmcli show controller` command to display details of the controller.

SYNTAX

```
dbadmcli show controller <controller_id> [-h]
```

PARAMETER

Parameter	Description
controller_id	The ID number of the controller. Use the `dbadmcli show storage` command to get the ID.
-h	(Optional) Displays help for using the command.

DBADMCLI SHOW DISK

Use the `dbadmcli show disk` command to display the status of a single disk or all disks on the DB system.

SYNTAX

```
dbadmcli show disk [<name>] [-shared] [-all] [-getlog] [-h]
```

PARAMETERS

Parameter	Description
-all	(Optional) Displays detailed information for the named disk.
-h	(Optional) Displays help for using the command.
-getlog	(Optional) Displays all the SMART log entries for an NVMe disk.
name	(Optional) Identifies a specific disk resource name. The resource name format is pd_[0..3]. If omitted, the command displays information about all disks on the system.
-shared	(Optional) Displays all the shared disks.

EXAMPLES

To display the status of all the disks on the system:

```
[root@dbsys ~]# dbadmcli show disk
       NAME            PATH            TYPE            STATE           STATE_DETAILS

       pd_00           /dev/nvme2n1    NVD             ONLINE          Good
       pd_01           /dev/nvme3n1    NVD             ONLINE          Good
       pd_02           /dev/nvme1n1    NVD             ONLINE          Good
       pd_03           /dev/nvme0n1    NVD             ONLINE          Good
```

To display the status of a disk named pd_00:

```
[root@dbsys ~]# dbadmcli show disk pd_00
The Resource is : pd_00
       ActionTimeout   :       1500
       ActivePath      :       /dev/nvme2n1
       AsmDiskList     :       |data_00||reco_00|
       AutoDiscovery   :       1
       AutoDiscoveryHi :       |data:70:NVD||reco:30:NVD|
       CheckInterval   :       300
       ColNum          :       0
       CriticalWarning :       0
       DependListOpr   :       add
       Dependency      :       |0|
       DiskId          :       360025380144d5332
       DiskType        :       NVD
       Enabled         :       1
       ExpNum          :       29
       HbaPortNum      :       10
       IState          :       0
       Initialized     :       0
       IsConfigDepende :       false
       ModelNum        :       MS1PC2DD3ORA3.2T
       MonitorFlag     :       1
       MultiPathList   :       |/dev/nvme2n1|
       Name            :       pd_00
       NewPartAddr     :       0
       OSUserType      :       |userType:Multiuser|
       PlatformName    :       X5_2_LITE_IAAS
       PrevState       :       Invalid
       PrevUsrDevName  :
       SectorSize      :       512
       SerialNum       :       S2LHNAAH502855
       Size            :       3200631791616
       SlotNum         :       0
```

```
       SmartDiskWarnin :       0
       SmartTemperatur :       32
       State           :       Online
       StateChangeTs   :       1467176081
       StateDetails    :       Good
       TotalSectors    :       6251233968
       TypeName        :       0
       UsrDevName      :       NVD_S00_S2LHNAAH502855
       VendorName      :       Samsung
       gid             :       0
       mode            :       660
       uid             :       0
```

To display the SMART logs for an NVMe disk:

```
[root@dbsys ~]# dbadmcli show disk pd_00 -getlog
SMART / Health Information :
----------------------------
Critical Warning : Available Spare below Threshold        : FALSE
Critical Warning : Temperature above Threshold            : FALSE
Critical Warning : Reliability Degraded                   : FALSE
Critical Warning : Read-Only Mode                         : FALSE
Critical Warning : Volatile Memory Backup Device Failure  : FALSE
Temperature                                               : 32 degree
Celsius
Available Spare                                           : 100%
Available Spare Threshold                                 : 10%
Device Life Used                                          : 0%
Data Units Read (in 512k byte data unit)                  : 89493
Data Units Written (in 512k byte data unit)               : 270387
Number of Host Read Commands                              : 4588381
Number of Host Write Commands                             : 6237344
Controller Busy Time                                      : 3 minutes
Number of Power Cycles                                    : 227
Number of Power On Hours                                  : 1115
Number of Unsafe Shutdowns                                : 218
Number of Media Errors                                    : 0
Number of Error Info Log Entries                          : 0
```

DBADMCLI SHOW DISKGROUP

Use the `dbadmcli show diskgroup` command to list configured diskgroups or display a specific diskgroup configuration.

SYNTAX

To list configured diskgroups:

```
dbadmcli show diskgroup [-h]
```

To display DATA configurations:

```
dbadmcli show diskgroup [DATA] [-h]
```

To display RECO configurations:

```
dbadmcli show diskgroup [RECO] [-h]
```

PARAMETERS

Parameter	Description
DATA	(Optional) Displays the DATA diskgroup configurations.
-h	(Optional) Displays help for using the command.
RECO	(Optional) Displays the RECO diskgroup configurations.

EXAMPLES

To list all diskgroups:

```
[root@dbsys ~]# dbadmcli show diskgroup

DiskGroups
----------
DATA
RECO
```

To display DATA configurations:

```
[root@dbsys ~]# dbadmcli show diskgroup DATA

ASM_DISK       PATH                            DISK    STATE   STATE_DETAILS
data_00        /dev/NVD_S00_S2LHNAAH101026p1   pd_00   ONLINE  Good
data_01        /dev/NVD_S01_S2LHNAAH101008p1   pd_01   ONLINE  Good
```

DBADMCLI SHOW ENV_HW

Use the `dbadmcli show env_hw` command to display the environment type and hardware version of the current DB system.

SYNTAX

```
dbadmcli show env_hw [-h]
```

PARAMETER

Parameter	Description
-h	(Optional) Displays help for using the command.

DBADMCLI SHOW FS

Use the `dbadmcli show fs` command to display file system details.

SYNTAX

```
dbadmcli show fs [-h]
```

PARAMETER

Parameter	Description
-h	(Optional) Displays help for using the command.

DBADMCLI SHOW STORAGE

Use the `dbadmcli show storage` command to show the storage controllers, expanders, and disks.

SYNTAX

```
dbadmcli show storage [-h]
```

To show storage errors:

```
dbadmcli show storage -errors [-h]
```

PARAMETERS

Parameter	Description
-errors	(Optional) Shows storage errors.
-h	(Optional) Displays help for using the command.

EXAMPLE

To display storage devices:

```
[root@dbsys ~]# dbadmcli show storage
==== BEGIN STORAGE DUMP ========
Host Description: Oracle Corporation:ORACLE SERVER X5-2
Total number of controllers: 5
        Id          = 4
        Pci Slot    = -1
        Serial Num  =
        Vendor      =
        Model       =
        FwVers      =
        strId       = iscsi_tcp:00:00.0
        Pci Address = 00:00.0

        Id          = 0
        Pci Slot    = 13
        Serial Num  = S2LHNAAH504431
        Vendor      = Samsung
        Model       = MS1PC2DD3ORA3.2T
        FwVers      = KPYA8R3Q
        strId       = nvme:25:00.00
        Pci Address = 25:00.0

        Id          = 1
        Pci Slot    = 12
        Serial Num  = S2LHNAAH505449
        Vendor      = Samsung
        Model       = MS1PC2DD3ORA3.2T
        FwVers      = KPYA8R3Q
        strId       = nvme:27:00.00
        Pci Address = 27:00.0
```

```
        Id          = 2
        Pci Slot    = 10
        Serial Num  = S2LHNAAH503573
        Vendor      = Samsung
        Model       = MS1PC2DD3ORA3.2T
        FwVers      = KPYA8R3Q
        strId       = nvme:29:00.00
        Pci Address = 29:00.0

        Id          = 3
        Pci Slot    = 11
        Serial Num  = S2LHNAAH503538
        Vendor      = Samsung
        Model       = MS1PC2DD3ORA3.2T
        FwVers      = KPYA8R3Q
        strId       = nvme:2b:00.00
        Pci Address = 2b:00.0

Total number of expanders: 0
Total number of PDs: 4
        /dev/nvme2n1    Samsung           NVD 3200gb slot:  0  pci : 29
        /dev/nvme3n1    Samsung           NVD 3200gb slot:  1  pci :  2
        /dev/nvme1n1    Samsung           NVD 3200gb slot:  2  pci : 27
        /dev/nvme0n1    Samsung           NVD 3200gb slot:  3  pci : 25
==== END STORAGE DUMP =========
```

DBADMCLI STORDIAG

Use the `dbadmcli stordiag` command to collect detailed information for each disk or NVM Express (NVMe).

SYNTAX

```
dbadmcli stordiag <name> [-h]
```

PARAMETERS

Parameter	Description
name	Defines the disk resource name. The resource name format is pd_[0..3].
-h	(Optional) Displays help for using the command.

EXAMPLE

To display detailed information for NVMe pd_00:

```
[root@dbsys ~]# dbadmcli stordiag pd_0
```

Database Sizing Templates

When you create a database using the `dbcli create-database` command, you can specify a database sizing template with the `--dbshape` parameter. The sizing templates are configured for different types of database workloads. Choose the template that best matches the most common workload your database performs:

- Use the OLTP templates if your database workload is primarily online transaction processing (OLTP).
- Use the DSS templates if your database workload is primarily decision support (DSS) or data warehousing.
- Use the in-memory (IMDB) templates if your database workload can fit in memory, and can benefit from in-memory performance capabilities.

The following tables describe the templates for each type of workload.

OLTP Database Sizing Templates

Template	CPU Cores	SGA (GB)	PGA (GB)	Flash (GB)	Processes	Redo Log File Size (GB)	Log Buffer (MB)
odb1s	1	2	1	6	200	1	16
odb1	1	4	2	12	200	1	16
odb2	2	8	4	24	400	1	16
odb4	4	16	8	48	800	1	32
odb6	6	24	12	72	1200	2	64
odb8	8	32	16	n/a	1600	2	64
odb10	10	40	20	n/a	2000	2	64
odb12	12	48	24	144	2400	4	64
odb16	16	64	32	192	3200	4	64
odb20	20	80	40	n/a	4000	4	64
odb24	24	96	48	192	4800	4	64
odb32	32	128	64	256	6400	4	64
odb36	36	128	64	256	7200	4	64

DSS Database Sizing Templates

Template	CPU Cores	SGA (GB)	PGA (GB)	Processes	Redo Log File Size (GB)	Log Buffer (MB)
odb1s	1	1	2	200	1	16
odb1	1	2	4	200	1	16
odb2	2	4	8	400	1	16
odb4	4	8	16	800	1	32
odb6	6	12	24	1200	2	64
odb8	8	16	32	1600	2	64
odb10	10	20	40	2000	2	64
odb12	12	24	48	2400	4	64
odb16	16	32	64	3200	4	64
odb20	20	40	80	4000	4	64
odb24	24	48	96	4800	4	64
odb32	32	64	128	6400	4	64
odb36	36	64	128	7200	4	64

IN-MEMORY DATABASE SIZING TEMPLATES

Template	CPU Cores	SGA (GB)	PGA (GB)	In-Memory (GB)	Processes	Redo Log Lile Size (GB)	Log Buffer (MB)
odb1s	1	2	1	1	200	1	16
odb1	1	4	2	2	200	1	16
odb2	2	8	4	4	400	1	16
odb4	4	16	8	8	800	1	32
odb6	6	24	12	12	1200	2	64
odb8	8	32	16	16	1600	2	64
odb10	10	40	20	20	2000	2	64
odb12	12	48	24	24	2400	4	64
odb16	16	64	32	32	3200	4	64
odb20	20	80	40	40	4000	4	64
odb24	24	96	48	48	4800	4	64
odb32	32	128	64	64	6400	4	64
odb36	36	128	64	64	7200	4	64

Overview of Autonomous Database

Oracle Cloud Infrastructure's Autonomous Database is a fully managed, preconfigured database environment with two workload types available, Autonomous Transaction Processing and Autonomous Data Warehouse. You do not need to configure or manage any hardware, or install any software. After provisioning, you can scale the number of CPU cores or the storage

capacity of the database at any time without impacting availability or performance. Autonomous Database handles creating the database, as well as the following maintenance tasks:

- Backing up the database
- Patching the database
- Upgrading the database
- Tuning the database

Available Workload Types

Autonomous Database offers two workload types:

- The **Autonomous Transaction Processing** workload type configures the database for a transactional workload, with a bias towards high volumes of random data access.
  For a complete product overview of Autonomous Transaction Processing, see Autonomous Transaction Processing. For Autonomous Transaction Processing tutorials, see Quick Start tutorials.
- The **Autonomous Data Warehouse** workload type configures the database for a decision support or data warehouse workload, with a bias towards large data scanning operations.
  For a complete product overview of Autonomous Data Warehouse, see Autonomous Data Warehouse. For Autonomous Data Warehouse tutorials, see Quick Start tutorials.

Availability

Both Autonomous Database products are currently available in the following regions:

Region Name	Region Location	Region Key
ap-seoul-1	Asia-Pacific: Seoul, South Korea	ICN
ap-tokyo-1	Asia-Pacific: Tokyo, Japan	NRT
ca-toronto-1	Canada: Toronto	YYZ
eu-frankfurt-1	Europe: Frankfurt, Germany	FRA
uk-london-1	United Kingdom: London	LHR
us-ashburn-1	United States: Ashburn, VA	IAD
us-phoenix-1	United States: Phoenix, AZ	PHX

Security Considerations

Autonomous Databases have public IP addresses and therefore are accessible to any client on the internet by default. Oracle recommends that you secure access to your database by using a service gateway or an access control list (ACL).

Service Gateway

Oracle Autonomous Database is one of the Oracle Cloud services that can be privately accessed through a service gateway within a virtual cloud network (VCN). This means you do not need a public IP or NAT to access your Autonomous Database instance from any of the cloud services within the Oracle Services Network. For example, if you have a Compute instance that uses a VCN with a service gateway, you can route traffic between your Compute instance and an Autonomous Database in the same region without the traffic going over the internet. For information on setting up a VCN service gateway and configuring it to access all supported Oracle Service Network services (which include Autonomous Database), see Access to Oracle Services: Service Gateway.

Access Control List (ACL)

An access control list provides additional protection for your Autonomous Database by allowing only the IP addresses in the list to connect to the database.

When you provision a new Autonomous Database, it does not have an initial ACL. You can use the Oracle Cloud Infrastructure Console, API, or CLI to create an ACL for the database by adding a minimum of one entry to the list. An entry can be a comma-separated list of CIDR blocks or public IP addresses. You can modify the list at any time. Removing all entries from the list makes the database accessible to all clients with the applicable credentials. See To manage the access control list of an Autonomous Database to learn how to create, update, or delete an ACL.



Important

If you are using a service gateway and you configure an access control list, you must add the CIDR range 240.0.0.0/4 to the ACL to enable clients accessing the database through the service gateway to connect to it.

Using the Oracle Cloud Infrastructure Console to Manage Autonomous Databases

For information on provisioning, managing, and backing up an Autonomous Database in the Oracle Cloud Infrastructure Console, see the following topics:

- Creating an Autonomous Database
- Managing an Autonomous Database
- Connecting to an Autonomous Database
- Backing Up an Autonomous Database Manually
- Restoring an Autonomous Database

Additional Autonomous Database Product Information

Autonomous Transaction Processing

For in-depth documentation on using and managing your Autonomous Transaction Processing database, see the following topics:

- Getting Started with Autonomous Transaction Processing
- Connecting to Autonomous Transaction Processing
- Loading Data with Autonomous Transaction Processing
- Querying External Data with Autonomous Transaction Processing
- Creating Dashboards, Reports, and Notebooks with Autonomous Transaction Processing
- Managing Users on Autonomous Transaction Processing
- Managing and Monitoring Performance of Autonomous Transaction Processing

For information on using a database client to manage your database, see the following topic:

- Connect Autonomous Transaction Processing Using a Client Application

Autonomous Data Warehouse

For in-depth documentation on using and managing your Autonomous Data Warehouse database, see the following topics:

- Getting Started with Autonomous Data Warehouse Cloud
- Connecting to Autonomous Data Warehouse Cloud
- Loading Data with Autonomous Data Warehouse Cloud
- Migrating Data from Amazon Redshift

- Querying External Data with Autonomous Data Warehouse Cloud
- Creating Dashboards, Reports, and Notebooks with Autonomous Data Warehouse Cloud
- Managing Users on Autonomous Data Warehouse Cloud
- Managing and Monitoring Performance of Autonomous Data Warehouse Cloud

For information on using a database client to manage your database, see the following topic:

- Connect Autonomous Data Warehouse Using a Client Application

Creating an Autonomous Database

This topic describes how to provision a new Autonomous Database using the Oracle Cloud Infrastructure Console or the API. For *Oracle By Example* tutorials on provisioning Autonomous Databases, see Provisioning Autonomous Transaction Processing and Provisioning Autonomous Data Warehouse Cloud.

Prerequisites

- To create an Autonomous Database, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in. See Authentication and Authorization for more information on user authorizations for the Oracle Cloud Infrastructure Database service. See Let database admins manage Autonomous Databases for sample Autonomous Database policies. See Details for the Database Service for detailed information on policy syntax.
- For information on additional prerequisites for provisioning an Autonomous Transaction Processing database, see What Do You Need? Likewise, for information on additional prerequisites for provisioning an Autonomous Data Warehouse, see What Do You Need?

Using the Oracle Cloud Infrastructure Console to Create an Autonomous Database

You can use the console to create either an Autonomous Database with either the Autonomous Transaction Processing or the Autonomous Data Warehouse workload type.

To Create an Autonomous Database

1. Open the navigation menu. Under **Database**, click **Autonomous Transaction Processing** or **Autonomous Data Warehouse**.
2. In the **Create Autonomous Database** dialog, enter the following:
    - **Workload Type:** Select the desired workload type. See About Autonomous Transaction Processing and About Autonomous Data Warehouse for information about each workload type.
    - **Compartment:** Select the compartment of the Autonomous Database.
    - **Display Name:** A user-friendly description or other information that helps you easily identify the resource. The display name does not have to be unique, and you can change it whenever you like. Avoid entering confidential information.
    - **Database Name:** The database name must consist of letters and numbers only, starting with a letter. The maximum length is 14 characters. Avoid entering confidential information.

      Note

      A database name cannot be used concurrently for both an Autonomous Data Warehouse and an Autonomous Transaction Processing database.

    - **CPU Core Count:** You can enable up to 128 cores for your Autonomous Database. The actual number of available cores is subject to your tenancy's service limits.

- **Storage:** Specify the storage you wish to make available to your Autonomous Database, in terabytes. You can make up to 128 TB available.
- **Administrator Credentials:** Set the password for the Autonomous Database Admin user by entering a password that meets the following criteria. You use this password when accessing the Autonomous Database service console and when using an SQL client tool.
    - Between 12 and 30 characters long
    - Contains at least one lowercase letter
    - Contains at least one uppercase letter
    - Contains at least one number
    - Does not contain the double quotation mark (")
    - Does not contain the string "admin", regardless of casing
- **License Type:** The type of license you want to use for the Autonomous Database. Your choice affects metering for billing. You have the following options:
    - **My Organization Already Owns Oracle Database Software Licenses**: This choice is used for the Bring Your Own License (BYOL) license type. If you choose this option, make sure you have proper entitlements to use for new service instances that you create.
    - **Subscribe to New Database Software Licenses and the Database Cloud Service**: This is used for the License Included license type. With this choice, the cost of the cloud service includes a license for the Database service.
- **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.Avoid entering confidential information.

3. Click **Create Autonomous Database**.

Using the API

Use the CreateAutonomousDatabase API operation to create Autonomous Transaction Processing databases of either the Transaction Processing or Warehouse workload types.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

For More Information

Autonomous Transaction Processing

- Using Oracle Autonomous Transaction Processing (the Autonomous Transaction Processing user guide)
- Autonomous Transaction Processing: Tutorials (Oracle By Example tutorials)
- Autonomous Transaction Processing: Videos (video tutorials)

Autonomous Data Warehouse

- Using Oracle Autonomous Data Warehouse (the Autonomous Data Warehouse user guide)
- Autonomous Data Warehouse: Tutorials (Oracle By Example tutorials)
- Autonomous Data Warehouse: Videos (video tutorials)

Connecting to an Autonomous Database

This topic gives an overview of connecting a client to an Autonomous Database and describes how to obtain the credentials and information you need to create a connection.

About Connecting to Autonomous Databases

Applications and tools connect to Autonomous Databases by using Oracle Net Services (also known as SQL*Net). SQL*Net supports a variety of connection types to Autonomous

Databases, including Oracle Call Interface (OCI), ODBC drivers, JDBC OC, and JDBC Thin Driver.

To support connections of any type, you'll need to download the client security credentials and network configuration settings required to access your database. You'll also need to supply the applicable TNS names or connection strings for a connection, depending on the client application or tool, type of connection, and service level. You can view or copy the TNS names and connection strings in the DB Connection dialog for your Autonomous Database. For detailed information about the TNS names, see Predefined Database Service Names for Autonomous Transaction Processing and Predefined Database Service Names for Autonomous Data Warehouse.

CONNECTING FROM A VCN

To connect to Autonomous Databases from a VCN, the VCN must be configured with one of the following gateways:

- internet gateway: For access from a public subnet in the VCN
- service gateway: For access from a private subnet in the VCN

Make sure to configure the subnet's route table with a rule that sends the desired traffic to the specific gateway. Also configure the subnet's security lists to allow the desired traffic.

About Downloading Client Credentials

The client credentials .zip that you download contains the following files:

- cwallet.sso - Oracle auto-login wallet
- ewallet.p12 - PKCS #12 wallet file associated with the auto-login wallet
- sqlnet.ora - SQL*Net profile configuration file that includes the wallet location and TNSNAMES naming method
- tnsnames.ora - SQL*Net configuration file that contains network service names mapped to connect descriptors for the local naming method
- Java Key Store (JKS) files - Key store files for use with JDBC Thin Connections



Important

Wallet files, along with the database user ID and password, provide access to data in your Autonomous Database. Store wallet files in a secure location. Share wallet files only with authorized users. If wallet files are transmitted in a way that might be accessed by unauthorized users (for example, over public email), transmit the wallet password separately and securely.

Before You Begin

The Autonomous Database is preconfigured to support Oracle Net Services (a TNS listener is installed and configured to use secure TCPS and client credentials.) The client computer must be prepared to use Oracle Net Services to connect to the Autonomous Database. Preparing your client includes downloading the client credentials. See the following links for steps you might have to perform before you access the client credentials and connection information for your Autonomous Database

Autonomous Transaction Processing

- Preparing for Oracle Call Interface (OCI), ODBC, and JDBC OCI Connections
- Preparing for JDBC Thin Connections

Autonomous Data Warehouse

- Preparing for Oracle Call Interface (OCI), ODBC, and JDBC OCI Connections
- Preparing for JDBC Thin Connections

Using the Oracle Cloud Infrastructure Console

To access the client credentials and connection information for your Autonomous Database

1. Open the navigation menu. Under **Database**, click **Autonomous Transaction Processing** or **Autonomous Data Warehouse**.
2. Choose your **Compartment**.
3. In the list of Autonomous Databases, click on the display name of the database you are interested in.
4. Click **DB Connection**.
5. To obtain the client credentials, click **Download**.
   You will be prompted to provide a password to encrypt the keys inside the wallet. The password must be at least 8 characters long and must include at least 1 letter and either 1 numeric character or 1 special character.
   Save the client credentials zip file to a secure location. See About Downloading Client Credentials for information about the files included in the download.
6. Take note of or copy the TNS names or connection strings you need for your connection. See About Connecting to Autonomous Databases for information about making connections.

Using the API

Use the GenerateAutonomousDatabaseWallet API operation to download the client credentials for your Autonomous Database.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

What's Next

For information and instructions on making secure connections to your database, see Connecting to Autonomous Transaction Processing and Connecting to Autonomous Data

Warehouse.

Managing an Autonomous Database

This topic describes the infrastructure management tasks for Autonomous Databases that you complete using the Oracle Cloud Infrastructure Console or the API. Note that some database management tasks not covered here are performed using the Autonomous Transaction Processing service console or the Autonomous Data Warehouse service console.

Prerequisites

To perform the management tasks in this topic, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in. See Authentication and Authorization for more information on user authorizations for the Oracle Cloud Infrastructure Database service. See Let database admins manage Autonomous Databases for sample Autonomous Database policies. See Details for the Database Service for detailed information on policy syntax.

Using the Oracle Cloud Infrastructure Console

You can perform basic administrative tasks for Autonomous Databases in the Oracle Cloud Infrastructure Console including stopping, starting, and scaling your databases. You can also use the Console to back up or to restore the database.

To set the Admin password

1. Open the navigation menu. Under **Database**, click **Autonomous Transaction Processing** or **Autonomous Data Warehouse**.
2. Choose your **Compartment**.

3. In the list of Autonomous Databases, click on the display name of the database you wish to administer.
4. Hover over the **Actions** button, and then click **Admin Password**. The Admin Password dialog opens.
5. Enter a password for the Autonomous Database. The password must meet the following criteria:
    - Between 12 and 30 characters long
    - Contains at least one lowercase letter
    - Contains at least one uppercase letter
    - Contains at least one number
    - Does not contain the double quotation mark (")
    - Does not contain the string "admin", regardless of casing
    - Is not one of the last four passwords used for the database
    - Is not a password you previously set within the last 24 hours
6. Enter the password again in the Confirm Password field.
7. Click **Update**.

To scale the CPU core count or storage of an Autonomous Database

1. Open the navigation menu. Under **Database**, click **Autonomous Transaction Processing** or **Autonomous Data Warehouse**.
2. Choose your **Compartment**.
3. In the list of Autonomous Databases, click on the display name of the database you wish to administer.
4. Click **Scale Up/Down**.
5. Enter a new value for **CPU Core Count** or **Storage** between 1 and 128. The number you enter represents the desired total (final) value for your database's CPU core count

or storage.

The number of available cores is subject to your tenancy's service limits. An Autonomous Database database can have a maximum of 128 cores and 128 TB of storage. Scaling the CPU core count affects your CPU billing.

6. Click **Update**.

To stop or start an Autonomous Database

1. Open the navigation menu. Under **Database**, click **Autonomous Transaction Processing** or **Autonomous Data Warehouse**.
2. Choose your **Compartment**.
3. In the list of Autonomous Databases, click on the display name of the database you wish to administer.
4. Hover over the **Actions** button, and then click **Stop** (or **Start**). When you stop your Autonomous Database, billing stops for CPU usage. Billing for storage continues when the database is stopped.
5. Confirm that you wish to stop or start your Autonomous Database in the confirmation dialog.



Note

Stopping your database has the following consequences:

- On-going transactions are rolled back.
- CPU billing is halted based on full-hour cycles of usage.
- You will not be able to connect to your database using database clients or tools.

To terminate an Autonomous Database



Warning

Terminating an Autonomous Database permanently deletes it. The database data, including automatic backups, will be lost when the system is terminated. Manual backups remain in Object Storage and are not automatically deleted when you terminate an Autonomous Database. Oracle recommends that you create a manual backup prior to terminating.

1. Open the navigation menu. Under **Database**, click **Autonomous Transaction Processing** or **Autonomous Data Warehouse**.
2. Choose your **Compartment**.
3. In the list of Autonomous Databases, click on the display name of the database you wish to administer.
4. Hover over the **Actions** button, and then click **Terminate**.
5. Confirm that you wish to terminate your Autonomous Database in the confirmation dialog.

To check the lifecycle state of your Autonomous Database

1. Open the navigation menu. Under **Database**, click **Autonomous Transaction Processing** or **Autonomous Data Warehouse**.
2. Choose your **Compartment**.
3. In the list of Autonomous Databases, click on the display name of the database you wish to administer.

4. In the Information tab, note the value displayed for **Lifecycle State**. For some lifecycle states, an information icon ( ) is displayed to provide additional details regarding the lifecycle state or ongoing operations such as backups, restores, or terminations. The database has one of the following lifecycle states:
    - Available
    - Available needs attention
    - Backup in progress
    - Provisioning
    - Restore in progress
    - Scaling in progress
    - Starting
    - Stopping
    - Stopped
    - Terminating
    - Terminated
    - Unavailable

To manage tags for your Autonomous Database

1. Open the navigation menu. Under **Database**, click **Autonomous Transaction Processing** or **Autonomous Data Warehouse**.
2. Choose your **Compartment**.
3. In the list of Autonomous Databases, click on the display name of the database you wish to administer.
4. Hover over the **Actions** button, and then click **Apply Tag(s)** to add new tags. Or click the **Tags** tab to view or edit the existing tags.

For more information, see Resource Tags.

To access the Autonomous Database Service Console

1. Open the navigation menu. Under **Database**, click **Autonomous Transaction Processing** or **Autonomous Data Warehouse**.
2. Choose your **Compartment**.
3. In the list of Autonomous Databases, click on the display name of the database you wish to administer.
4. Click **Service Console**.

For information on using the Autonomous Transaction Processing service console features, see Managing and Monitoring Performance of Autonomous Transaction Processing. For information on using the Autonomous Data Warehouse service console features, see Managing and Monitoring Performance of Autonomous Data Warehouse Cloud.

To change the license type of an Autonomous Database

1. Open the navigation menu. Under **Database**, click **Autonomous Transaction Processing** or **Autonomous Data Warehouse**.
2. Choose your **Compartment**.
3. In the list of Autonomous Databases, click on the display name of the database you wish to administer.
4. Hover over the **Actions** button, and then click **Update License Type**.
   The dialog displays the options with your current license type selected.
5. Select the new license type.
6. Click **Update**.

See Known Issue.

To manage the access control list of an Autonomous Database

A newly-provisioned Autonomous Database does not have an access control list. Without one, the database can be accessed by any client with the applicable credentials to connect to it. To create an ACL, you must add at least one CIDR block or public IP address to the list.

1. Open the navigation menu. Under **Database**, click **Autonomous Transaction Processing** or **Autonomous Data Warehouse**.
2. Choose your **Compartment**.
3. In the list of Autonomous Databases, click on the display name of the database you wish to administer.
4. Hover over the **Actions** button, and then click **Access Control List**.
5. Add or modify entries, as applicable.
   To add CIDR blocks or public IP addresses of clients that can access the database, click **+ Additional Entry**, and then select the IP notation type and enter a comma-separated list of values.

   Important

   If you are using a service gateway, ensure that the CIDR range 240.0.0.0/4 is included in the list to allow clients accessing the database through the service gateway to connect to it.

   To remove the ACL, simply delete all entries in the list. This action allows all clients to connect to the database.
6. Click **Update**.

For information about access control lists, see Access Control List (ACL).

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage Autonomous Databases:

- ListAutonomousDatabases
- GetAutonomousDatabase
- UpdateAutonomousDatabase
- StartAutonomousDatabase
- StopAutonomousDatabase
- DeleteAutonomousDatabase

For More Information

Autonomous Transaction Processing

- Managing Users on Autonomous Transaction Processing
- Managing and Monitoring Performance of Autonomous Transaction Processing
- Autonomous Transaction Processing Quickstart Tutorials
- Autonomous Transaction Processing (complete user guide)

Autonomous Data Warehouse

- Managing Users on Autonomous Data Warehouse Cloud
- Managing and Monitoring Performance of Autonomous Data Warehouse Cloud
- Autonomous Data Warehouse Quickstart Tutorials.
- Autonomous Data Warehouse (complete user guide)

Backing Up an Autonomous Database Manually

This topic describes how to create manual backups of Autonomous Databases. You can use the Oracle Cloud Infrastructure Console or the API to perform these tasks.

Oracle Cloud Infrastructure automatically backs up your Autonomous Databases and retains these backups for 60 days. Automatic backups are weekly full backups and daily incremental backups. You can also create manual backups to supplement your automatic backups. Manual backups are stored in an Object Storage bucket that you create, and are retained for 60 days.



Note

During the backup operation, the database remains available. However, lifecycle management operations such as stopping the database, scaling it, or terminating it are disabled.

Prerequisites

- To create or manage Autonomous Database backups, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in. See Authentication and Authorization for more information on user authorizations for the Oracle Cloud Infrastructure Database service. See Let database admins manage Autonomous Databases for sample Autonomous Database policies. See Details for the Database Service for detailed information on policy syntax.
- To create a manual backup for an Autonomous Database, you must first configure an Object Storage bucket to serve as a destination for your manual backups. See Setting Up a Bucket to Store Manual Backups for instructions.

Using the Oracle Cloud Infrastructure Console

1. Open the navigation menu. Under **Database**, click **Autonomous Transaction Processing** or **Autonomous Data Warehouse**.
2. Choose your **Compartment**.
3. In the list of Autonomous Databases, find the database that you want to back up.
4. Click the name to display the Autonomous Database details.
5. Click **Create Manual Backup**.



Tip

If this step does not successfully complete, confirm that you have an Object Storage bucket set up and configured to store manual backups. See Setting Up a Bucket to Store Manual Backups for instructions.

6. In the **Create Manual Backup** dialog, enter a name for your backup. Avoid entering confidential information.
7. Click **Create**. Your backup may take several hours to complete, depending on the size of your database.
8. Optionally, you can check the state of your backup in the list of backups on the database details page. For some states, an information icon ( ) is displayed to provide additional details regarding the state or ongoing operations like deletions. The backup has one of the following states:
   - Creating
   - Active
   - Deleting
   - Deleted
   - Failed

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage Autonomous Database backups:

- ListAutonomousDatabaseBackups
- GetAutonomousDatabaseBackup
- CreateAutonomousDatabaseBackup

Setting Up a Bucket to Store Manual Backups

You must create an Oracle Cloud Infrastructure Object Storage bucket to hold your Autonomous Database manual backups and configure your database to connect to it. This is a one-time operation.

To set up an object store and user credentials for your manual backups

Some of the steps in this procedure require you to connect to the database by using an Oracle Database client such as SQL Developer. See Connecting with Oracle SQL Developer (18.2 or later) for information and instructions on connecting to an Autonomous Transaction Processing database. See Connecting with Oracle SQL Developer (18.2 or later) for information and instructions on connecting to an Autonomous Data Warehouse database.

1. If you have not already done so, generate an auth token for the Oracle Cloud Infrastructure Object Storage user to access the bucket you create in the next step. See To create an auth token to learn how to do this. (You will need this auth token for the database credential you create in step 4.)
2. In the Oracle Cloud Infrastructure Console, create a bucket in your designated Object Storage Swift compartment to hold the backups. The format of the bucket name is `backup_databasename`, where *databasename* is lowercase.

For example, if you provision a database named DATABASE1, the bucket name should be `backup_database1`.



Note

When you create your bucket:

- Pick **Standard** as the storage tier. Manual backups are only supported with buckets created in the standard storage tier.
- Ensure that you use the database name, and not the display name, as the bucket name.

3. Using an Oracle Database client, log in to the database as the administrator set the database `default_bucket` property to your Oracle Cloud Infrastructure Object Storage tenancy URL. The format of the tenancy URL is https://swiftobjectstorage.*region*.oraclecloud.com/v1/*object_storage_namespace*. For example:

```
ALTER DATABASE PROPERTY SET default_bucket='https://swiftobjectstorage.us-phoenix-
1.oraclecloud.com/v1/ansh8lvru1zp';
```

In the example, the Object Storage namespace is **ansh8lvru1zp**.



Tip

- To determine the Object Storage namespace string to use, click **View Bucket** in the actions (three dots) menu of the bucket you created in the previous step.
- Do not include the bucket name in the URL.
- Ensure that you follow the format indicated. Do not use a pre-authenticated request URL.

4. With the tenancy user and the auth token referenced in step 1, create the credential for your Oracle Cloud Infrastructure Object Storage account. Use `DBMS_CLOUD.CREATE_CREDENTIAL` to create the credential.

   For example:

   ```
   BEGIN
   DBMS_CLOUD.CREATE_CREDENTIAL(
   credential_name => 'DEF_CRED_NAME',
   username => 'db1_user@oracle.com',
   password => '<auth_token>'
   );
   END;
   /
   ```

   For more information on creating this credential, see CREATE_CREDENTIAL Procedure.

5. Set the database property `default_credential` to the credential you created in the previous step.

   For example:

   ```
   ALTER DATABASE PROPERTY SET default_credential = 'ADMIN.DEF_CRED_NAME';
   ```

To list the current value for the default bucket, run the following command:

```
SELECT PROPERTY_VALUE from database_properties WHERE PROPERTY_NAME='DEFAULT_BUCKET';
```

After completing these steps you can take manual backups any time you want.

Restoring an Autonomous Database

This topic describes how to restore an Autonomous Database from a backup. You can use the Oracle Cloud Infrastructure Console or the API to perform this task.

You can use any existing manual or automatic backup to restore your database, or you can restore and recover your database to any point in time in the 60-day retention period of your automatic backups. For point-in-time restores, you specify a timestamp, and your Autonomous Database decides which backup to use for the fastest restore.



Note

Restoring Autonomous Database puts the database in the unavailable state during the restore operation. You cannot connect to a database in that state. The only lifecycle management operation supported in the unavailable state is terminate.

Prerequisites

To restore Autonomous Databases, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in. See Authentication and Authorization for more information on user authorizations for the Oracle Cloud Infrastructure Database service. See Let database admins manage Autonomous Databases for sample Autonomous Database policies. See Details for the Database Service for detailed information on policy syntax.

Using the Oracle Cloud Infrastructure Console

To restore an Autonomous Database from a backup

1. Open the navigation menu. Under **Database**, click **Autonomous Transaction Processing** or **Autonomous Data Warehouse**.
2. Choose your **Compartment**.
3. In the list of Autonomous Databases, find the database that you wish to restore.
4. Click the name of the Autonomous Database to display the database details.
5. Click the **Restore** button to open the restore dialog.
6. Click **Select Backup**.
7. Specify the date range for a list of backups to display.
8. Select the backup.
9. Click **Restore**.

To restore an Autonomous Database using point-in-time restore

1. Open the navigation menu. Under **Database**, click **Autonomous Transaction Processing** or **Autonomous Data Warehouse**.
2. Choose your **Compartment**.
3. In the list of Autonomous Databases, find the database that you wish to restore.
4. Click the name of the Autonomous Database to display the database details.
5. Click the **Restore** button to open the restore dialog.
6. Click **Specify Timestamp**.
7. Enter a timestamp. Your Autonomous Database decides which backup to use for faster recovery. The timestamp input allows you to specify precision to the seconds level (YYYY-MM-DD HH:MM:SS GMT).

8. Click **Restore**.

Using the API

Use the RestoreAutonomousDatabase API operation to restore your Autonomous Database from a backup.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Cloning an Autonomous Database

This topic describes how to clone an existing Autonomous Database using the Oracle Cloud Infrastructure Console or the API. You may wish to use the cloning feature to create a point-in-time copy of your Autonomous Database for purposes such as testing, development or analytics. If you need to clone only the database schema of your source database, the Console's cloning feature is a quick and easy way to accomplish this task.

Clone Types

The clone feature offers the following two types of Autonomous Database clones:

- The full clone option creates a new database that includes all of the source database's metadata and data.
- The metadata clone option creates a new database that includes the source database's metadata, but not the source database's data.

Prerequisites

- To clone an Autonomous Database, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the

type of access you've been granted and which compartment you should work in. See Authentication and Authorization for more information on user authorizations for the Oracle Cloud Infrastructure Database service.

Using the Oracle Cloud Infrastructure Console

To clone an Autonomous Database

1. Open the navigation menu. Under **Database**, click **Autonomous Transaction Processing** or **Autonomous Data Warehouse**.
2. Choose your **Compartment**.
3. In the list of Autonomous Databases, click on the display name of the database you wish to clone.
4. Hover over the **Actions** button, and then click **Create Clone**.
   In the **Create Autonomous Database Clone** dialog, enter the following:

   Clone Type

   Select the type of clone you wish to create. Choose either **Full Clone** or **Metadata Clone**.

   Database Information

   - **Compartment:**Your current compartment is the default selection.
   - **Display Name:** A user-friendly description or other information that helps you easily identify the resource. The display name does not have to be unique, and you can change it whenever you like. Avoid entering confidential information.
   - **Database Name:** The database name must consist of letters and numbers only, starting with a letter. The maximum length is 14 characters. Avoid entering confidential information.
   - **CPU Core Count:** You can enable up to 128 cores for your Autonomous Database. The actual number of available cores is subject to your tenancy's service limits.

- **Storage:** Specify the storage you wish to make available to your Autonomous Database database, in terabytes. You can make up to 128 TB available. For full clones, the size of the source database determines the minimum amount of storage you can make available.

ADMINISTRATOR CREDENTIAL

Set the password for the Autonomous Database Admin user by entering a password that meets the following criteria. You use this password when accessing the Autonomous Database service console and when using an SQL client tool.

- Between 12 and 30 characters long
- Contains at least one lowercase letter
- Contains at least one uppercase letter
- Contains at least one number
- Does not contain the double quotation mark (")
- Does not contain the string "admin", regardless of casing

LICENSE TYPE

The type of license you want to use for the Autonomous Transaction Processing database. Your choice affects metering for billing. You have the following options:

- **My Organization Already Owns Oracle Database Software Licenses**: This choice is used for the Bring Your Own License (BYOL) license type. If you choose this option, make sure you have proper entitlements to use for new service instances that you create.
- **Subscribe to New Database Software Licenses and the Database Cloud Service**: This is used for the License Included license type. With this choice, the cost of the cloud service includes a license for the Database service.

5. Click **Create Autonomous Database Clone**.

The Console displays the details page for the new clone of your database and the service begins provisioning the Autonomous Database. Note the following:

- The new clone displays the **Provisioning** lifecycle state until the provisioning process completes.
- The source database remains in the **Available** lifecycle state.
- Backups associated with the source database are not cloned for either the full clone or the metadata clone option.
- Oracle recommends that you evaluate the security requirements for the new database and implement them, as applicable. See Security Considerations for details.

Using the API

Use the CreateAutonomousDatabase API operation to clone an Autonomous Database.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

For More Information

For information about optimizer statistics, resource management rules and performance data for a cloned database, see the Using Oracle Autonomous Data Warehouse and Using Oracle Autonomous Transaction Processing user guides.

Migrating Databases to the Cloud

You can migrate your on-premises Oracle Database to an Oracle Cloud Infrastructure Database service database using a number of different methods that use several different tools. The method that applies to a given migration scenario depends on several factors, including the version, character set, and platform endian format of the source and target databases.

Choosing a Migration Method

Not all migration methods apply to all migration scenarios. Many of the migration methods apply only if specific characteristics of the source and destination databases match or are

compatible. Moreover, additional factors can affect which method you choose for your migration from among the methods that are technically applicable to your migration scenario.

Some of the characteristics and factors to consider when choosing a migration method are:

- On-premises database version
- Database service database version
- On-premises host operating system and version
- On-premises database character set
- Quantity of data, including indexes
- Data types used in the on-premises database
- Storage for data staging
- Acceptable length of system outage
- Network bandwidth

To determine which migration methods are applicable to your migration scenario, gather the following information.

1. Database version of your on-premises database:
    - Oracle Database 12c Release 2 version 12.2.0.1
    - Oracle Database 12c Release 1 version 12.1.0.2 or higher
    - Oracle Database 12c Release 1 version lower than 12.1.0.2
    - Oracle Database 11g Release 2 version 11.2.0.3 or higher
    - Oracle Database 11g Release 2 version lower than 11.2.0.3
2. For on-premises Oracle Database 12c Release 2 and Oracle Database 12c Release 1 databases, the architecture of the database:
    - Multitenant container database (CDB)
    - Non-CDB
3. Endian format (byte ordering) of your on-premises database's host platform

Some platforms are little endian and others are big endian. Query V$TRANSPORTABLE_PLATFORM to identify the endian format, and to determine whether cross-platform tablespace transport is supported.

The Oracle Cloud Infrastructure Database uses the Linux platform, which is little endian.

4. Database character set of your on-premises database and the Oracle Cloud Infrastructure Database database.

   Some migration methods require that the source and target databases use compatible database character sets.

5. Database version of the Oracle Cloud Infrastructure Database database you are migrating to:

   - Oracle Database 12c Release 2
   - Oracle Database 12c Release 1
   - Oracle Database 11g Release 2

   Oracle Database 12c Release 2 and Oracle Database 12c Release 1 databases created on the Database service use CDB architecture. Databases created using the Enterprise Edition software edition are single-tenant, and databases created using the High Performance or Extreme Performance software editions are multitenant.

After gathering this information, use the "source" and "destination" database versions as your guide to see which migration methods apply to your migration scenario:

- Migrating from Oracle Database 11g to Oracle Database 11g in the Cloud
- Migrating from Oracle Database 11g to Oracle Database 12c in the Cloud
- Migrating from Oracle Database 12c CDB to Oracle Database 12c in the Cloud
- Migrating from Oracle Database 12c Non-CDB to Oracle Database 12c in the Cloud

Migration Connectivity Options

You have several connectivity options when migrating your on-premises databases to the Oracle Cloud Infrastructure. The options are listed below in order of preference.

1. **FastConnect:** Provides a secure connection between your existing network and your virtual cloud network (VCN) over a private physical network instead of the internet. For more information, see FastConnect.
2. **IPSec VPN:** Provides a secure connection between a dynamic routing gateway (DRG) and customer-premise equipment (CPE), consisting of multiple IPSec tunnels. The IPSec connection is one of the components forming a site-to-site VPN between a VCN and your on-premises network. For more information, see VPN Connect.
3. **Internet gateway:** Provides a path for network traffic between your VCN and the internet. For more information, see Internet Gateway.

Migration Methods

Many methods exist to migrate Oracle databases to the Oracle Cloud Infrastructure Database service. Which of these methods apply to a given migration scenario depends on several factors, including the version, character set, and platform endian format of the source and target databases.

- Data Pump Conventional Export/Import
- Data Pump Full Transportable
- Data Pump Transportable Tablespace
- Remote Cloning a PDB
- Remote Cloning Non-CDB
- RMAN Cross-Platform Transportable PDB
- RMAN Cross-Platform Transportable Tablespace Backup Sets
- RMAN Transportable Tablespace with Data Pump
- RMAN DUPLICATE from an Active Database
- RMAN CONVERT Transportable Tablespace with Data Pump
- SQL Developer and INSERT Statements to Migrate Selected Objects
- SQL Developer and SQL*Loader to Migrate Selected Objects

- Unplugging/Plugging a PDB
- Unplugging/Plugging Non-CDB

Migrating an On-Premises Database to Oracle Cloud Infrastructure by Creating a Backup in the Cloud



Note

This topic is not applicable to Exadata DB systems.

You can migrate an on-premises database to Oracle Cloud Infrastructure by creating a backup of your on-premises database in Oracle Cloud Infrastructure's Database service.

Oracle provides a Python script to create a backup of your database. The script invokes an API call to create the backup and then places the backup in Oracle Cloud Infrastructure. You can then use the Console or the API to create a new database or DB system from that backup. Backups created using the instructions in this topic appear under Standalone Backups in the console.

The Python script is bundled as a part of the Oracle Cloud Infrastructure CLI installation. Oracle provides the migration script and associated files at no cost. Normal Object Storage charges apply for the storage of your backup in Oracle Cloud Infrastructure.

Compatibility

The scripted migration process is compatible with the following bare metal and virtual machine DB system configurations:

Configuration	Version or Type	Notes
Database Version	18.x 12.2.0.1 12.1.0.2 11.2.0.4	• For versions 18c, 12.2.0.1, and 12.1.0.2: ◦ Only Container Databases (CDBs) are supported. The scripted migration process may work with non-CDB databases for these database versions, but Oracle does not provide support for the migration of non-CDB databases using the script described in this topic. For information on creating an on-premises pluggable database (PDB) by cloning a non-CDB in Oracle Database 18c, see About Cloning a Non-CDB. For an overview of multitenant architecture in Oracle Database 18c, see Introduction to the Multitenant Architecture. For information on creating an on-premises pluggable database (PDB) from a non-CDB database in Oracle Database 12c Release 2 (12.2), see Upgrading a Non-CDB Oracle Database To a PDB on a CDB. For an overview of multitenant architecture in 12c Release 2, see Overview of Managing a Multitenant Environment. ◦ The Oracle Cloud Infrastructure Database service will attempt to run datapatch, which requires read/write mode. If there are pluggable databases (PDBs), they should also be in read/write mode to ensure that datapatch runs on them. • For version 11.2.0.4, depending on the source

Configuration	Version or Type	Notes
		database patch level, you may need to roll back patches prior to migrating. See Rolling Back Patches on a Version 11.2 Database for more information. • If your on-premises database has an interim patch (previous known as a one-off patch), see Applying Interim Patches for details on applying the patch in Oracle Cloud Infrastructure.
Source Database Platform	Oracle Enterprise Linux / Red Hat Enterprise Linux 5.x Oracle Linux / Red Hat Enterprise Linux 6.x Oracle Linux / Red Hat Enterprise 7.x	• The scripted migration described in this topic may work in Microsoft Windows environments, but Oracle currently does not provide support for this script in Windows. • For Oracle Linux 6.x users, see Configuring Oracle Linux 6 to install Python for details on configuring the operating system to install a compatible version of Python. See Installing the CLI for more information regarding Oracle Linux 6.
Encryption	TDE Non-TDE	• In a non-TDE configuration, the RMAN encryption password is required. • Unencrypted on-premises databases remain unencrypted when restored to Oracle Cloud Infrastructure. The stored RMAN standalone backups are always encrypted.

Configuration	Version or Type	Notes
Target Database Edition	Standard Edition Enterprise Edition Enterprise Edition - High Performance Enterprise Edition - Extreme Performance	
Cluster	Single RAC	

Prerequisites

On the source database host:

- Outbound internet connectivity for installing Python packages, running yum install, and access to the Oracle Cloud Infrastructure API and Object Storage.
- RMAN configuration to autobackup `controlfile` and `spfile`:

```
RMAN> CONFIGURE CONTROLFILE AUTOBACKUP ON;
```



Note

RMAN configuration changes must be completed prior to running the script. The script may modify RMAN parameters as required to complete the backup and migration tasks.

To Migrate an On-Premises Database Using a Standalone Backup

Perform the following tasks on the source database host:

1. Create a directory named `/home/oracle/migrate`.



Tip

You can name the `migrate` portion of the directory path anything you want. If you use a different name, you must adjust all of the paths that appear in this task accordingly. The following examples assume the name `migrate` for simplicity and clarity.

2. As *root*, run the CLI installer in the directory you created in step 1. (For example, `/home/oracle/migrate`.) See Installing the CLI: Windows for instructions on running the installer script in Windows. See Installing the CLI: MacOS, Linux, and Unix for instructions on running the installer script in the Bash environment.

   The installer installs Python 3.6.0 if either Python 2.7 or Python 3.6 does not exist on the machine. The installer also installs the Python script required to create and migrate a standalone backup from an on-premises database.

   On Oracle Linux 6, a newer version of Python (such as Python 3.6.0) is usually required. **Use the following instructions to configure Oracle Linux 6 before running the backup script.**

Configuring Oracle Linux 6 to install Python

In Oracle Linux 6 use the following `/etc/yum.repos.d/ol6.repo` file to ensure that a compatible version of Python is installed by the script if a compatible version is not already installed. Include this file before attempting to run the script with the `./install.sh` command.

```
[ol6_latest]
name=Oracle Linux $releasever Latest ($basearch)
baseurl=http://yum.oracle.com/repo/OracleLinux/OL6/latest/$basearch/
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-oracle
gpgcheck=1
enabled=1
```

3. Copy the following files into the new directory:
   - Oracle Database Backup Module (`opc_install.jar`)
   - Your API `*.pem` key file.
4. Respond to the prompts as follows:

```
(yum install)
Is this ok [y/N]: y

===> Missing native dependencies. Continue and install the following dependencies: gcc, libffi-
devel, python36u-devel, openssl-devel? (Y/n): Y

===> In what directory would you like to place the install? (leave blank to use
'/root/lib/oracle-cli'): /home/oracle/migrate/lib/oracle-cli

===> In what directory would you like to place the 'oci' executable? (leave blank to use
'/root/bin'): /home/oracle/migrate/bin

===> In what directory would you like to place the OCI scripts? (leave blank to use
'/root/bin/oci-cli-scripts'): /home/oracle/migrate/bin/oci-cli-scripts

===> Currently supported optional packages are: ['db (will install cx_Oracle)'] What optional CLI
packages would you like to be installed (comma separated names; press enter if you don't need any
optional packages)?: db
```

```
===> Modify profile to update your $PATH and enable shell/tab completion now? (Y/n): Y

===> Enter a path to an rc file to update (leave blank to use '/root/.bashrc'):
/home/oracle/.bashrc
```

5. Perform the following file operations:

```
# chown -R oracle:oinstall /home/oracle/migrate
```

6. Edit the `/home/oracle/migrate/config.txt` file

```
[DEFAULT]
tenancy=<your_tenancy_OCID>
user=<your_user_OCID>
fingerprint=<fingerprint>
key_file=/home/oracle/migrate/<your_api_key>.pem
region=<region>
```

If you do not know your API signing key's fingerprint, see How to Get the Key's Fingerprint.

7. As *oracle user* (not *root*), run one of the following sets of commands, depending on the type of database you are migrating.

For a non-TDE database:

```
export AD=<destination_availability_domain>
export C=<destination_compartment_OCID>
export ORACLE_SID=<ORACLE_SID>
export ORACLE_HOME=<ORACLE_HOME>
export PATH=$PATH:$ORACLE_HOME/bin
export LC_ALL=en_US.UTF-8
export ORACLE_UNQNAME=<source_DB_unique_name>
rm -rf /home/oracle/migrate/onprem_upload
cd /home/oracle/migrate/bin/oci-cli-scripts/
./create_backup_from_onprem --config-file /home/oracle/migrate/config.txt --display-name
<example_display_name> --availability-domain $AD --edition ENTERPRISE_EDITION_EXTREME_PERFORMANCE
--opc-installer-dir /home/oracle/migrate --tmp-dir /home/oracle/migrate/onprem_upload --
compartment-id $C --rman-password <password>
```

For a TDE-enabled database:

```
export AD=<destination_availability_domain>
export C=<destination_compartment_OCID>
export ORACLE_SID=<ORACLE_SID>
```

```
export ORACLE_HOME=<ORACLE_HOME>
export PATH=$PATH:$ORACLE_HOME/bin
rm -rf /home/oracle/migrate/onprem_upload
cd /home/oracle/migrate/bin/oci-cli-scripts/
./create_backup_from_onprem --config-file /home/oracle/migrate/config.txt --display-name
<example_display_name> --availability-domain $AD --edition ENTERPRISE_EDITION_EXTREME_PERFORMANCE
--opc-installer-dir /home/oracle/migrate --tmp-dir /home/oracle/migrate/onprem_upload --
compartment-id $C
```

See the following list of parameters used by the script for more details.

Parameters used by the script

Parameter	Description	Required
`--config-file`	The path to the oci-cli config file. The default path is as follows: `~/.oci/config`	No
`--profile`	The profile in the config file to load. This profile will also be used to locate any default parameter values which have been specified in the OCI CLI-specific configuration file. The default value is `DEFAULT`.	No
`--compartment-id`	The compartment OCID of the Oracle Cloud Infrastructure compartment that will contain your standalone backup.	Yes
`--display-name`	The name of the backup, as you wish it to be displayed in the OCI Console under Standalone Backups.	Yes
`--availability-domain`	The availability domain where the backup is to be stored.	Yes

Parameter	Description	Required
`--edition`	The edition of the Oracle Cloud Infrastructure DB system that will contain the database created from the standalone backup. You can choose the same edition as the on-premises database, or any addition above the on-premises database. The choices, listed from lowest to highest, are the following: • STANDARD_EDITION • ENTERPRISE_EDITION • ENTERPRISE_EDITION_HIGH_PERFORMANCE • ENTERPRISE_EDITION_EXTREME_PERFORMANCE	Yes
`--opc-installer-dir`	The directory containing the `opc_installer.jar` file. This is the directory you created in step 1 of this procedure.	Yes
`--additional-opc-args`	Optional additional arguments for the opc installer.	No
`--tmp-dir`	Optional temporary directory for intermediate files.	No
`--rman-password`	The RMAN password to use for the standalone backup.	Required if TDE is not enabled

Parameter	Description	Required
`--rman-channels`	RMAN channels. The default value is 5.	No
`--help`	Displays in-line help for the script in the OCI-CLI environment.	No

The script will produce a standalone backup of your on-premises database in your Oracle Cloud Infrastructure tenancy. You can check the Console for your backup by viewing the **Standalone Backups** page in the Database service, under **Bare Metal, VM, and Exadata**.



Tip

To access command line help for the backup script, run the following command in the /home/oracle/migrate/bin/oci-cli-scripts/ directory:

```
create_backup_from_onprem --help
```

8. Create a new database or launch a new DB system using the backup you created in the preceding step. See the following instructions to perform these tasks:
   - Recovering a Database from Object Storage
   - Recovering a Database from Object Storage

Migrating from Oracle Database 11g to Oracle Database 11g in the Cloud

You can migrate Oracle Database 11g databases from on-premises to Oracle Database 11g databases in the Database service using several different methods.

The applicability of some of the migration methods depends on the on-premises database's character set and platform endian format.

If you have not already done so, determine the database character set of your on-premises database, and determine the endian format of the platform your on-premises database resides on. Use this information to help you choose an appropriate method.

- Data Pump Conventional Export/Import
  This method can be used regardless of the endian format and database character set of the on-premises database.
  For the steps this method entails, see Data Pump Conventional Export/Import.
- Data Pump Transportable Tablespace
  This method can be used only if the on-premises platform is little endian, and the database character sets of your on-premises database and the Oracle Cloud Infrastructure Database database are compatible.
  For the steps this method entails, see Data Pump Transportable Tablespace.
- RMAN Transportable Tablespace with Data Pump
  This method can be used only if the on-premises platform is little endian, and the database character sets of your on-premises database and the Oracle Cloud Infrastructure Database database are compatible.
  For the steps this method entails, see RMAN Transportable Tablespace with Data Pump.
- RMAN `CONVERT` Transportable Tablespace with Data Pump
  This method can be used only if the database character sets of your on-premises database and the Oracle Cloud Infrastructure Database database are compatible.
  This method is similar to the Data Pump Transportable Tablespace method, with the addition of the RMAN `CONVERT` command to enable transport between platforms with different endianness. Query `V$TRANSPORTABLE_PLATFORM` to determine if the on-premises database platform supports cross-platform tablespace transport and to determine the endian format of the platform. The Database service platform is little-endian format.
  For the steps this method entails, see RMAN CONVERT Transportable Tablespace with Data Pump.

Migrating from Oracle Database 11g to Oracle Database 12c in the Cloud

You can migrate Oracle Database 11g databases from on-premises to Oracle Database 12c databases in the Database service using several different methods.

The applicability of some of the migration methods depends on the on-premises database's version, database character set and platform endian format.

If you have not already done so, determine the database version and database character set of your on-premises database, and determine the endian format of the platform your on-premises database resides on. Use this information to help you choose an appropriate method.

- Data Pump Conventional Export/Import
  This method can be used regardless of the endian format and database character set of the on-premises database.
  For the steps this method entails, see Data Pump Conventional Export/Import.
- Data Pump Transportable Tablespace
  This method can be used only if the on-premises platform is little endian, and the database character sets of your on-premises database and the Databaseservice database are compatible.
  For the steps this method entails, see Data Pump Transportable Tablespace.
- RMAN Transportable Tablespace with Data Pump
  This method can be used only if the on-premises platform is little endian, and the database character sets of your on-premises database and the Database service database are compatible.
  For the steps this method entails, see RMAN Transportable Tablespace with Data Pump.
- RMAN `CONVERT` Transportable Tablespace with Data Pump
  This method can be used only if the database character sets of your on-premises database and the Database service database are compatible.
  This method is similar to the Data Pump Transportable Tablespace method, with the addition of the RMAN `CONVERT` command to enable transport between platforms with different endianness. Query `V$TRANSPORTABLE_PLATFORM` to determine if the on-

premises database platform supports cross-platform tablespace transport and to determine the endian format of the platform. The Database service platform is little-endian format.

For the steps this method entails, see RMAN CONVERT Transportable Tablespace with Data Pump.

- Data Pump Full Transportable

  This method can be used only if the source database release version is 11.2.0.3 or later, and the database character sets of your on-premises database and the Database service database are compatible.

  For the steps this method entails, see Data Pump Full Transportable.

Migrating from Oracle Database 12c CDB to Oracle Database 12c in the Cloud

You can migrate Oracle Database 12c CDB databases from on-premises to Oracle Database 12c databases in the Oracle Cloud Infrastructure Database service using several different methods.

The applicability of some of the migration methods depends on the on-premises database's character set and platform endian format.

If you have not already done so, determine the database character set of your on-premises database, and determine the endian format of the platform your on-premises database resides on. Use this information to help you choose an appropriate method.

- Data Pump Conventional Export/Import

  This method can be used regardless of the endian format and database character set of the on-premises database.

  For the steps this method entails, see Data Pump Conventional Export/Import.

- Data Pump Transportable Tablespace

  This method can be used only if the on-premises platform is little endian, and the database character sets of your on-premises database and the Database database are compatible.

For the steps this method entails, see Data Pump Transportable Tablespace.

- RMAN Transportable Tablespace with Data Pump
  This method can be used only if the on-premises platform is little endian, and the database character sets of your on-premises database and the Oracle Cloud Infrastructure Database service database are compatible.
  For the steps this method entails, see RMAN Transportable Tablespace with Data Pump.
- RMAN `CONVERT` Transportable Tablespace with Data Pump
  This method can be used only if the database character sets of your on-premises database and the Database database are compatible.
  This method is similar to the Data Pump Transportable Tablespace method, with the addition of the RMAN `CONVERT` command to enable transport between platforms with different endianness. Query `V$TRANSPORTABLE_PLATFORM` to determine if the on-premises database platform supports cross-platform tablespace transport and to determine the endian format of the platform. The Database service platform is little-endian format.
  For the steps this method entails, see RMAN CONVERT Transportable Tablespace with Data Pump.
- RMAN Cross-Platform Transportable Tablespace Backup Sets
  This method can be used only if the database character sets of your on-premises database and the Database service database are compatible.
  For the steps this method entails, see RMAN Cross-Platform Transportable Tablespace Backup Sets.
- Data Pump Full Transportable
  This method can be used only if the database character sets of your on-premises database and the Database service database are compatible.
  For the steps this method entails, see Data Pump Full Transportable.
- Unplugging/Plugging (CDB)
  This method can be used only if the on-premises platform is little endian, and the on-premises database and Database database have compatible database character sets and national character sets.

For the steps this method entails, see Unplugging/Plugging a PDB.

- Remote Cloning (CDB)
  This method can be used only if the on-premises platform is little endian, the on-premises database release is 12.1.0.2 or higher, and the on-premises database and Database service database have compatible database character sets and national character sets.
  For the steps this method entails, see Remote Cloning a PDB.
- RMAN Cross-Platform Transportable PDB
  This method can be used only if the on-premises platform is little endian, and the database character sets of your on-premises database and the Database service database are compatible.
  For the steps this method entails, see RMAN Cross-Platform Transportable PDB.
- SQL Developer and SQL*Loader to Migrate Selected Objects
  You can use SQL Developer to create a cart into which you add selected objects to be loaded into your Oracle Database 12c database on the cloud. In this method, you use SQL*Loader to load the data into your cloud database.
  For the steps this method entails, see SQL Developer and SQL*Loader to Migrate Selected Objects.
- SQL Developer and `INSERT` Statements to Migrate Selected Objects
  You can use SQL Developer to create a cart into which you add selected objects to be loaded into your Oracle Database 12c database on the cloud. In this method, you use SQL `INSERT` statements to load the data into your cloud database.
  For the steps this method entails, see SQL Developer and INSERT Statements to Migrate Selected Objects.

Migrating from Oracle Database 12c Non-CDB to Oracle Database 12c in the Cloud

You can migrate Oracle Database 12c non-CDB databases from on-premises to Oracle Database 12c databases in Oracle Cloud Infrastructure Database service using several different methods.

The applicability of some of the migration methods depends on the on-premises database's character set and platform endian format.

If you have not already done so, determine the database character set of your on-premises database, and determine the endian format of the platform your on-premises database resides on. Use this information to help you choose an appropriate method.

- Data Pump Conventional Export/Import
  This method can be used regardless of the endian format and database character set of the on-premises database.
  For the steps this method entails, see Data Pump Conventional Export/Import.
- Data Pump Transportable Tablespace
  This method can be used only if the on-premises platform is little endian, and the database character sets of your on-premises database and the Database database are compatible.
  For the steps this method entails, see Data Pump Transportable Tablespace.
- RMAN Transportable Tablespace with Data Pump
  This method can be used only if the on-premises platform is little endian, and the database character sets of your on-premises database and the Database service database are compatible.
  For the steps this method entails, see RMAN Transportable Tablespace with Data Pump.
- RMAN `CONVERT` Transportable Tablespace with Data Pump
  This method can be used only if the database character sets of your on-premises database and the Database service database are compatible.
  This method is similar to the Data Pump Transportable Tablespace method, with the addition of the RMAN `CONVERT` command to enable transport between platforms with different endianness. Query `V$TRANSPORTABLE_PLATFORM` to determine if the on-premises database platform supports cross-platform tablespace transport and to determine the endian format of the platform. The Database service platform is little-endian format.
  For the steps this method entails, see RMAN CONVERT Transportable Tablespace with Data Pump.

- RMAN Cross-Platform Transportable Tablespace Backup Sets
  This method can be used only if the database character sets of your on-premises database and the Database database are compatible.
  For the steps this method entails, see RMAN Cross-Platform Transportable Tablespace Backup Sets.
- Data Pump Full Transportable
  This method can be used only if the database character sets of your on-premises database and the Database service database are compatible.
  For the steps this method entails, see Data Pump Full Transportable.
- Unplugging/Plugging (non-CDB)
  This method can be used only if the on-premises platform is little endian, and the on-premises database and Database service database have compatible database character sets and national character sets.
  You can use the unplug/plug method to migrate an Oracle Database 12c non-CDB database to Oracle Database 12c in the cloud. This method provides a way to consolidate several non-CDB databases into a single Oracle Database 12c CDB on the cloud.
  For the steps this method entails, see Unplugging/Plugging Non-CDB.
- Remote Cloning (non-CDB)
  This method can be used only if the on-premises platform is little endian, the on-premises database release is 12.1.0.2 or higher, and the on-premises database and Database service database have compatible database character sets and national character sets.
  You can use the remote cloning method to copy an Oracle Database 12c non-CDB on-premises database to your Oracle Database 12c database in the cloud.
  For the steps this method entails, see Remote Cloning Non-CDB.
- SQL Developer and SQL*Loader to Migrate Selected Objects
  You can use SQL Developer to create a cart into which you add selected objects to be loaded into your Oracle Database 12c database on the cloud. In this method, you use SQL*Loader to load the data into your cloud database.

For the steps this method entails, see SQL Developer and SQL*Loader to Migrate Selected Objects.

- SQL Developer and `INSERT` Statements to Migrate Selected Objects
  You can use SQL Developer to create a cart into which you add selected objects to be loaded into your Oracle Database 12c database on the cloud. In this method, you use SQL `INSERT` statements to load the data into your cloud database.
  For the steps this method entails, see SQL Developer and INSERT Statements to Migrate Selected Objects.

Data Pump Conventional Export/Import

You can use this method regardless of the endian format and database character set of the on-premises database.

To migrate an on-premises source database, tablespace, schema, or table to the database on a Database service database deployment using Data Pump Export and Import, you perform these tasks:

1. On the on-premises database host, invoke Data Pump Export and export the on-premises database.
2. Use a secure copy utility to transfer the dump file to the Database service compute node.
3. On the Database service compute node, invoke Data Pump Import and import the data into the database.
4. After verifying that the data has been imported successfully, you can delete the dump file.

For information about Data Pump Import and Export, see these topics:

- "Data Pump Export Modes" in *Oracle Database Utilities* for Release 12.2, 12.1 or 11.2.
- "Data Pump Import Modes" in *Oracle Database Utilities* for Release 12.2, 12.1 or 11.2.

Data Pump Conventional Export/Import: Example

This example provides a step-by-step demonstration of the tasks required to migrate a schema from an on-premises Oracle database to a Database service database.

This example illustrates a schema mode export and import. The same general procedure applies for a full database, tablespace, or table export and import.

In this example, the on-premises database is on a Linux host.

1. On the on-premises database host, invoke Data Pump Export to export the schemas.
    a. On the on-premises database host, create an operating system directory to use for the on-premises database export files.

    ```
    $ mkdir /u01/app/oracle/admin/orcl/dpdump/for_cloud
    ```

    b. On the on-premises database host, invoke SQL*Plus and log in to the on-premises database as the `SYSTEM` user.

    ```
    $ sqlplus system
    Enter password: <enter the password for the SYSTEM user>
    ```

    c. Create a directory object in the on-premises database to reference the operating system directory.

    ```
    SQL> CREATE DIRECTORY dp_for_cloud AS '/u01/app/oracle/admin/orcl/dpdump/for_cloud';
    ```

    d. Exit from SQL*Plus.
    e. On the on-premises database host, invoke Data Pump Export as the `SYSTEM` user or another user with the `DATAPUMP_EXP_FULL_DATABASE` role and export the on-premises schemas. Provide the password for the user when prompted.

    ```
    $ expdp system SCHEMAS=fsowner DIRECTORY=dp_for_cloud
    ```

2. Use a secure copy utility to transfer the dump file to the Database service compute node.
    In this example the dump file is copied to the `/u01` directory. Choose the appropriate location based on the size of the file that will be transferred.

a. On the Database service compute node, create a directory for the dump file.

```
$ mkdir /u01/app/oracle/admin/ORCL/dpdump/from_onprem
```

b. Before using the `scp` command to copy the export dump file, make sure the SSH private key that provides access to the Database service compute node is available on your on-premises host.

c. On the on-premises database host, use the SCP utility to transfer the dump file to the Databaseservice compute node.

```
$ scp –i private_key_file \
/u01/app/oracle/admin/orcl/dpdump/for_cloud/expdat.dmp \
oracle@IP_address_DBaaS_VM:/u01/app/oracle/admin/ORCL/dpdump/from_onprem
```

3. On the Database service compute node, invoke Data Pump Import and import the data into the database.

a. On the Database service compute node, invoke SQL*Plus and log in to the database as the `SYSTEM` user.

```
$ sqlplus system
Enter password: <enter the password for the SYSTEM user>
```

b. Create a directory object in the Database service database.

```
SQL> CREATE DIRECTORY dp_from_onprem AS '/u01/app/oracle/admin/ORCL/dpdump/from_onprem';
```

c. If they do not exist, create the tablespace(s) for the objects that will be imported.

d. Exit from SQL*Plus.

e. On the Database service compute node, invoke Data Pump Import and connect to the database. Import the data into the database.

```
impdp system SCHEMAS=fsowner DIRECTORY=dp_from_onprem
```

4. After verifying that the data has been imported successfully, you can delete the `expdat.dmp` file.

Data Pump Full Transportable

You can use this method only if the source database release version is 11.2.0.3 or later, and the database character sets of your on-premises database and the Oracle Cloud Infrastructure

Database service database are compatible.

You can use the Data Pump full transportable method to copy an entire database from your on-premises host to the database on a Database service database deployment.

To migrate an Oracle Database 11g on-premises database to the Oracle Database 12c database on a Database service database deployment using the Data Pump full transportable method, you perform these tasks:

1. On the on-premises database host, prepare the database for the Data Pump full transportable export by placing the user-defined tablespaces in `READ ONLY` mode.
2. On the on-premises database host, invoke Data Pump Export to perform the full transportable export.
3. Use a secure copy utility to transfer the Data Pump Export dump file and the datafiles for all of the user-defined tablespaces to the Database service compute node.
4. Set the on-premises tablespaces back to `READ WRITE`.
5. On the Database service compute node, prepare the database for the tablespace import.
6. On the Database service compute node, invoke Data Pump Import and connect to the database.
7. After verifying that the data has been imported successfully, you can delete the dump file.

Data Pump Full Transportable: Example

This example provides a step-by-step demonstration of the tasks required to migrate an Oracle Database 11g database to a Database service 12c database.

In this example, the source database is on a Linux host.

1. On the source database host, prepare the database for the Data Pump full transportable export.
    a. On the source database host, create a directory in the operating system to use for the source export.

```
$ mkdir /u01/app/oracle/admin/orcl/dpdump/for_cloud
```

b. On the source database host, invoke SQL*Plus and log in to the source database as the `SYSTEM` user.

```
$ sqlplus system
Enter password: <enter the password for the SYSTEM user>
```

c. Create a directory object in the source database to reference the operating system directory.

```
SQL> CREATE DIRECTORY dp_for_cloud AS '/u01/app/oracle/admin/orcl/dpdump/for_cloud';
```

d. Determine the name(s) of the tablespaces and data files that belong to the user-defined tablespaces by querying `DBA_DATA_FILES`. These files will also be listed in the export output.

```
SQL> SELECT tablespace_name, file_name FROM dba_data_files;
TABLESPACE_NAME  FILE_NAME
---------------  --------------------------------------------------
USERS            /u01/app/oracle/oradata/orcl/users01.dbf
UNDOTBS1         /u01/app/oracle/oradata/orcl/undotbs01.dbf
SYSAUX           /u01/app/oracle/oradata/orcl/sysaux01.dbf
SYSTEM           /u01/app/oracle/oradata/orcl/system01.dbf
EXAMPLE          /u01/app/oracle/oradata/orcl/example01.dbf
FSDATA           /u01/app/oracle/oradata/orcl/fsdata01.dbf
FSINDEX          /u01/app/oracle/oradata/orcl/fsindex01.dbf
SQL>
```

e. On the source database host, set all tablespaces that will be transported (the transportable set) to `READ ONLY` mode.

```
SQL> ALTER TABLESPACE example READ ONLY;
Tablespace altered.
SQL> ALTER TABLESPACE fsindex READ ONLY;
Tablespace altered.
SQL> ALTER TABLESPACE fsdata READ ONLY;
Tablespace altered.
SQL> ALTER TABLESPACE users READ ONLY;
Tablespace altered.
SQL>
```

f. Exit from SQL*Plus.

2. On the source database host, invoke Data Pump Export to perform the full transportable export. Specify `FULL=y` and `TRANSPORTABLE=always`. Because this is an Oracle Database 11g database and full transportable is an Oracle Database 12c feature, specify `VERSION=12`. Provide the password for the `SYSTEM` user when prompted.

```
$ expdp system FULL=y TRANSPORTABLE=always VERSION=12 DUMPFILE=expdat.dmp DIRECTORY=dp_for_cloud
```

3. Use a secure copy utility to transfer the Data Pump Export dump file and the datafiles for all of the user-defined tablespaces to the Database service compute node.
   In this example the dump file is copied to the `/u01` directory. Choose the appropriate location based on the size of the file that will be transferred.

   a. On the Database service compute node, create a directory for the dump file.

```
$ mkdir /u01/app/oracle/admin/ORCL/dpdump/from_source
```

   b. Before using the `scp` utility to copy files, make sure the SSH private key that provides access to the Database service compute node is available on your source host.

   c. On the source database host, use the `scp` utility to transfer the dump file and all datafiles of the transportable set to the Database service compute node.

```
$ scp -i private_key_file \
/u01/app/oracle/admin/orcl/dpdump/for_cloud/expdat.dmp \
oracle@compute_node_IP_address:/u01/app/oracle/admin/ORCL/dpdump/from_source

$ scp -i private_key_file \
/u01/app/oracle/oradata/orcl/example01.dbf \
oracle@compute_node_IP_address:/u02/app/oracle/oradata/ORCL/PDB2

$ scp -i private_key_file \
/u01/app/oracle/oradata/orcl/fsdata01.dbf \
oracle@compute_node_IP_address:/u02/app/oracle/oradata/ORCL/PDB2

$ scp -i private_key_file \
/u01/app/oracle/oradata/orcl/fsindex01.dbf \
oracle@compute_node_IP_address:/u02/app/oracle/oradata/ORCL/PDB2
```

```
$ scp -i private_key_file \
/u01/app/oracle/oradata/orcl/users01.dbf \
oracle@compute_node_IP_address:/u02/app/oracle/oradata/ORCL/PDB2
```

4. Set the source tablespaces back to `READ WRITE`.
    a. Invoke SQL*Plus and log in as the `SYSTEM` user.
    b. Set the user-defined tablespaces back to `READ WRITE` mode.

```
SQL> ALTER TABLESPACE example READ WRITE;
Tablespace altered.
SQL> ALTER TABLESPACE fsdata READ WRITE;
Tablespace altered.
SQL> ALTER TABLESPACE fsindex READ WRITE;
Tablespace altered.
SQL> ALTER TABLESPACE users READ WRITE;
Tablespace altered.
```

    c. Exit from SQL*Plus.
5. On the Database service compute node, prepare the PDB for the tablespace import.
    a. On the Database service compute node, invoke SQL*Plus and log in to the PDB as the `SYSTEM` user.
    b. Create a directory object in the PDB.

```
SQL> CREATE DIRECTORY dp_from_source AS '/u01/app/oracle/admin/ORCL/dpdump/from_source';
```

6. On the Database service compute node, invoke Data Pump Import and connect to the PDB.

    Import the data into the database using the `TRANSPORT_DATAFILES` option.

```
$ impdp system@PDB2 FULL=y DIRECTORY=dp_from_source \
TRANSPORT_
DATAFILES='/u02/app/oracle/oradata/ORCL/PDB2/example01.dbf',\
'/u02/app/oracle/oradata/ORCL/PDB2/fsdata01.dbf',\
'/u02/app/oracle/oradata/ORCL/PDB2/fsindex01.dbf,'\
'/u02/app/oracle/oradata/ORCL/PDB2/users01.dbf'
```

7. After verifying that the data has been imported successfully, you can delete the `expdat.dmp` dump file.

Data Pump Transportable Tablespace

You can use this method only if the on-premises platform is little endian, and the database character sets of your on-premises database and the Oracle Cloud Infrastructure Database service database are compatible.

The Transportable Tablespace method is generally much faster than a conventional export/import of the same data because the data files containing all of the actual data are simply copied to the destination location. You use Data Pump to transfer only the metadata of the tablespace objects to the new database.

To migrate an on-premises source database to the database deployment on the Database service using the Data Pump Transportable Tablespace method, you perform these tasks:

1. On the on-premises database host, prepare the database for the Data Pump transportable tablespace export.
2. On the on-premises database host, invoke Data Pump Export to perform the transportable tablespace export.
3. Use a secure copy utility to transfer the Data Pump Export dump file and the tablespace datafiles to the Database service compute node.
4. Set the on-premises tablespaces back to `READ WRITE`.
5. On the Databaseservice compute node, prepare the database for the tablespace import.
6. On the Database service compute node, invoke Data Pump Import and connect to the database.
7. Set the tablespaces on the Database service database to `READ WRITE` mode.
8. After verifying that the data has been imported successfully, you can delete the dump file.

Data Pump Transportable Tablespace: Example

This example provides a step-by-step demonstration of the tasks required to migrate tablespaces in an on-premises Oracle database to a Database service database.

This example performs a migration of the `FSDATA` and `FSINDEX` tablespaces.

In this example, the on-premises database is on a Linux host.

1. On the on-premises database host, prepare the database for the Data Pump transportable tablespace export.
    a. On the on-premises database host, create a directory in the operating system to use for the on-premises export.

```
mkdir /u01/app/oracle/admin/orcl/dpdump/for_cloud
```

    b. On the on-premises database host, invoke SQL*Plus and log in to the on-premises database as the `SYSTEM` user.

```
sqlplus system
Enter password: <enter the password for the SYSTEM user>
```

    c. Create a directory object in the on-premises database to reference the operating system directory.

```
SQL> CREATE DIRECTORY dp_for_cloud AS '/u01/app/oracle/admin/orcl/dpdump/for_cloud';
```

    d. Determine the name(s) of the datafiles that belong to the `FSDATA` and `FSINDEX` tablespaces by querying DBA_DATA_FILES. These files will also be listed in the export output.

```
SQL> SELECT file_name FROM dba_data_files
  2  WHERE tablespace_name = 'FSDATA';

FILE_NAME
----------------------------------------------------------------
/u01/app/oracle/oradata/orcl/fsdata01.dbf

SQL> SELECT file_name FROM dba_data_files
  2  WHERE tablespace_name = 'FSINDEX';

FILE_NAME
----------------------------------------------------------------
/u01/app/oracle/oradata/orcl/fsindex01.dbf
```

    e. On the on-premises database host, set all tablespaces that will be transported (the transportable set) to `READ ONLY` mode.

```
SQL> ALTER TABLESPACE fsindex READ ONLY;
Tablespace altered.
SQL> ALTER TABLESPACE fsdata READ ONLY;
Tablespace altered.
```

f. Exit from SQL*Plus.

2. On the on-premises database host, invoke Data Pump Export to perform the transportable tablespace export.

   On the on-premises database host, invoke Data Pump Export and connect to the on-premises database. Export the on-premises tablespaces using the `TRANSPORT_TABLESPACES` option. Provide the password for the `SYSTEM` user when prompted.

```
expdp system TRANSPORT_TABLESPACES=fsdata,fsindex TRANSPORT_FULL_CHECK=YES DIRECTORY=dp_for_cloud
```

3. Use a secure copy utility to transfer the Data Pump Export dump file and the tablespace datafiles to the Database service compute node.

   In this example the dump file is copied to the `/u01` directory. Choose the appropriate location based on the size of the file that will be transferred.

   a. On the Database service compute node, create a directory for the dump file.

```
mkdir /u01/app/oracle/admin/ORCL/dpdump/from_onprem
```

   b. Before using the `scp` utility to copy files, make sure the SSH private key that provides access to the Database service compute node is available on your on-premises host.

   c. On the on-premises database host, use the `scp` utility to transfer the dump file and all datafiles of the transportable set to the Database service compute node.

```
scp -i private_key_file \
/u01/app/oracle/admin/orcl/dpdump/for_cloud/expdat.dmp \
oracle@IP_address_DBaaS_VM:/u01/app/oracle/admin/ORCL/dpdump/from_onprem

$ scp -i private_key_file \/u01/app/oracle/oradata/orcl/fsdata01.dbf \
oracle@IP_address_DBaaS_VM:/u02/app/oracle/oradata/ORCL

$ scp -i private_key_file \/u01/app/oracle/oradata/orcl/fsindex01.dbf \
oracle@IP_address_DBaaS_VM:/u02/app/oracle/oradata/ORCL
```

4. Set the on-premises tablespaces back to `READ WRITE`.

a. Invoke SQL*Plus and log in as the `SYSTEM` user.

b. Set the `FSDATA` and `FSINDEX` tablespaces back to `READ WRITE` mode.

```
SQL> ALTER TABLESPACE fsdata READ WRITE;
Tablespace altered.
SQL> ALTER TABLESPACE fsindex READ WRITE;
Tablespace altered.
```

c. Exit from SQL*Plus.

5. On the Database service compute node, prepare the database for the tablespace import.

a. On the Database service compute node, invoke SQL*Plus and log in to the database as the `SYSTEM` user.

b. Create a directory object in the Database service database.

```
SQL> CREATE DIRECTORY dp_from_onprem AS '/u01/app/oracle/admin/ORCL/dpdump/from_onprem';
```

c. If the owners of the objects that will be imported do not exist in the database, create them before performing the import. The transportable tablespace mode of import does not create the users.

```
SQL> CREATE USER fsowner
  2  PROFILE default
  3  IDENTIFIED BY fspass
  4  TEMPORARY TABLESPACE temp
  5  ACCOUNT UNLOCK;
```

6. On the Database service compute node, invoke Data Pump Import and connect to the database.

   Import the data into the database using the `TRANSPORT_DATAFILES` option.

```
impdp system DIRECTORY=dp_from_onprem \
TRANSPORT_DATAFILES='/u02/app/oracle/oradata/ORCL/fsdata01.dbf', \
'/u02/app/oracle/oradata/ORCL/fsindex01.dbf'
```

7. Set the tablespaces on the Database service database to `READ WRITE` mode.

a. Invoke SQL*Plus and log in as the `SYSTEM` user.

b. Set the `FSDATA` and `FSINDEX` tablespaces to `READ WRITE` mode.

```
SQL> ALTER TABLESPACE fsdata READ WRITE;
Tablespace altered.
SQL> ALTER TABLESPACE fsindex READ WRITE;
Tablespace altered.
```

c. Exit from SQL*Plus.

8. After verifying that the data has been imported successfully, you can delete the `expdat.dmp` dump file.

Remote Cloning a PDB

You can use this method only if the on-premises platform is little endian, the on-premises database release is 12.1.0.2 or higher, and the on-premises database and Database service database have compatible database character sets and national character sets.

You can use the remote cloning method to copy a PDB from your on-premises Oracle Database 12c database to a PDB in an Oracle Database 12c database on the Database service.

Migration Tasks

To migrate an Oracle Database 12c PDB to a PDB in a Database service database deployment using the remote cloning method, you perform these tasks:

1. On the on-premises database host, invoke SQL*Plus and close the on-premises PDB and then reopen it in `READ ONLY` mode.
2. On the Database service compute node, invoke SQL*Plus and create a database link that enables a connection to the on-premises database.
3. On the Database service compute node, execute the `CREATE PLUGGABLE DATABASE` command to clone the on-premises PDB.
4. On the Database compute node, open the new PDB by executing the `ALTER PLUGGABLE DATABASE OPEN` command.
5. Optionally, on the on-premises database host invoke SQL*Plus and set the on-premises PDB back to `READ WRITE` mode.

For more information, see "Cloning a Remote PDB or Non-CDB" in *Oracle Database Administrator's Guide* for Release 12.2 or 12.1.

Remote Cloning Non-CDB

You can use this method only if the on-premises platform is little endian, the on-premises database release is 12.1.0.2 or higher, and the on-premises database and Database service database have compatible database character sets and national character sets.

You can use the remote cloning method to copy an Oracle Database 12c non-CDB on-premises database to a PDB in an Oracle Database 12c database on the Databaseservice.

Migration Tasks

To migrate an Oracle Database 12c non-CDB database to a Database service database deployment using the remote cloning method, you perform these tasks:

1. On the on-premises database host, invoke SQL*Plus and set the on-premises database to `READ ONLY` mode.
2. On the Database service compute node, invoke SQL*Plus and create a database link that enables a connection to the on-premises database.
3. On the Database service compute node, execute the `CREATE PLUGGABLE DATABASE` command to clone the on-premises non-CDB database.
4. On the Database service compute node, execute the `$ORACLE_HOME/rdbms/admin/noncdb_to_pdb.sql` script.
5. On the Database service compute node, open the new PDB by executing the `ALTER PLUGGABLE DATABASE OPEN` command.
6. Optionally, on the on-premises database host invoke SQL*Plus and set the on-premises database back to `READ WRITE` mode.

For more information, see "Cloning a Remote PDB or Non-CDB" in *Oracle Database Administrator's Guide* for Release 12.2 or 12.1.

RMAN Cross-Platform Transportable PDB

This method can be used only if the on-premises platform is little endian, and the database character sets of your on-premises database and the Database service database are compatible.

To migrate an Oracle Database 12c PDB to a PDB in an Oracle Database 12c database on a Database service deployment using the RMAN cross-platform transportable PDB method, you perform these tasks:

1. On the on-premises database host, invoke SQL*Plus and close the on-premises PDB.
2. On the on-premises database host, execute the `ALTER PLUGGABLE DATABASE UNPLUG` command to generate an XML file containing the list of datafiles that will be plugged in on the cloud database.
3. On the on-premises database host, invoke RMAN and connect to the root. Execute the `BACKUP FOR TRANSPORT PLUGGABLE DATABASE` command.
4. Use a secure copy utility to transfer the XML file and the backup set to the Database service compute node.
5. On the Database service compute node, invoke RMAN and connect to the root. Execute the `RESTORE ALL FOREIGN DATAFILES` command.
6. the Database service compute node, invoke SQL*Plus and connect to the root. Execute the `CREATE PLUGGABLE DATABASE` command.
7. the Database service compute node, execute the `ALTER PLUGGABLE DATABASE OPEN` command.

For more information, see " Performing Cross-Platform Data Transport in CDBs and PDBs" in *Oracle Database Backup and Recovery User's Guide* for Release 12.2 or 12.1.

RMAN Cross-Platform Transportable Tablespace Backup Sets

You can use this method only if the database character sets of your on-premises database and the Database service database are compatible.



Note

For detailed information on a similar method that enables you to perform a cross-platform transport of an entire database, see the *Oracle Database 12c Backup and Recovery User's Guide* for Release 12.2 or 12.1 . When you transport an entire database to a different platform, the source platform and the destination platform must use the same endian format.

To migrate Oracle Database 12c on-premises tablespaces to an Oracle Database 12c database on a Database service deployment using the RMAN cross-platform transportable backup sets method, you perform these tasks:

1. On the on-premises database host, prepare the database by placing the user-defined tablespaces that you intend to transport in `READ ONLY` mode.
2. On the on-premises database host, invoke RMAN and use the `BACKUP` command with the `TO PLATFORM` or `FOR TRANSPORT` clause and the `DATAPUMP` clause to create a backup set for cross-platform transport. See in "BACKUP" in *Oracle Database Backup and Recovery Reference* for Release 12.2 or 12.1 for more information on the `BACKUP` command.
3. Use a secure copy utility to transfer the backup sets, including the Data Pump export dump file, to the Database service compute node.
4. Set the on-premises tablespaces back to `READ WRITE`.
5. On the Database service compute node, prepare the database by creating the required schemas.
6. On the Database service compute node, invoke RMAN and use the `RESTORE` command with the `foreignFileSpec` subclause to restore the cross-platform backup.
7. On the Database service compute node, set the tablespaces on the database to `READ WRITE` mode.

For more information, see "Overview of Cross-Platform Data Transport Using Backup Sets" in *Oracle Database Backup and Recovery User's Guide* for Release 12.2 or 12.1.

RMAN Cross-Platform Transportable Tablespace Backup Sets: Example

This example provides a step-by-step demonstration of the tasks required to migrate tablespaces in an Oracle Database PDB to a Database service database.

This example performs a migration of the `FSDATA` and `FSINDEX` tablespaces.

In this example, the on-premises database is on a Linux host.

1. On the on-premises database host, prepare the database by creating a directory for the export dump file and placing the user-defined tablespaces that you intend to transport in READ ONLY mode..
    a. On the on-premises database host, create a directory in the operating system to use for the export dump.

    ```
    mkdir /u01/app/oracle/admin/orcl/dpdump/for_cloud
    ```

    b. On the on-premises data host, invoke SQL*Plus and log in to the PDB as the `SYSTEM` user..

    ```
    sqlplus system@pdb_servicename
    Enter password: enter the password for the SYSTEM user
    ```

    c. Create a directory object in the on-premises database to reference the operating system directory.

    ```
    SQL> CREATE DIRECTORY dp_for_cloud AS '/u01/app/oracle/admin/orcl/dpdump/for_cloud';
    ```

    d. On the on-premises database host, set all tablespaces that will be transported (the transportable set) to READ ONLY mode.

    ```
    SQL> ALTER TABLESPACE fsindex READ ONLY;
    SQL> ALTER TABLESPACE fsdata READ ONLY;
    ```

    e. Exit from SQL*Plus.
2. On the on-premises database host, invoke RMAN and use the `BACKUP` command with the `TO PLATFORM` or `FOR TRANSPORT` clause and the `DATAPUMP` clause to create a backup set for cross-platform transport.
    a. On the on-premises database host, create an operating system directory for the datafiles.

```
mkdir /u01/app/oracle/admin/orcl/rman_transdest
```

b. Invoke RMAN and log in as a user that has been granted the `SYSDBA` or `SYSBACKUP` privilege.

```
rman target username@pdb_servicename
```

c. Execute the `BACKUP` command.

```
RMAN> BACKUP FOR TRANSPORT
2> FORMAT '/u01/app/oracle/admin/orcl/rman_transdest/fs_tbs.bck'
3> TABLESPACE fsdata,fsindex
4> DATAPUMP FORMAT '/u01/app/oracle/admin/orcl/rman_transdest/fs_tbs.dmp';
```

d. Log out of RMAN.

e. Optionally, navigate to the directory you specified in the `BACKUP` command to view the files that were created.

```
cd /u01/app/oracle/admin/orcl/rman_transdest
$ ls
fs_tbs.bck  fs_tbs.dmp
```

3. Use a secure copy utility to transfer the backup set, including the Data Pump export dump file, to the Database service compute node.

a. On the Database service compute node, create a directory for the backup set and dump file.

```
mkdir /tmp/from_onprem
```

b. Before using the `scp` command to copy files, make sure the SSH private key that provides access to the Database service compute node is available on your on-premises host.

c. On the on-premises database host, use the SCP utility to transfer the backup set and the dump file to the Database service compute node.

```
scp -i private_key_file \
/u01/app/oracle/admin/orcl/rman_transdest/fs_tbs.bck \
oracle@IP_address_DBaaS_VM:/tmp/from_onprem

$ scp -i private_key_file \
/u01/app/oracle/admin/orcl/rman_transdest/fs_tbs.dmp \
```

```
oracle@IP_address_DBaaS_VM:/tmp/from_onprem

$
```

4. Set the on-premises tablespaces back to READ WRITE.
    a. Invoke SQL*Plus and log in to the PDB as the `SYSTEM` user.
    b. Set the `FSDATA` and `FSINDEX` tablespaces back to READ WRITE mode.

```
SQL> ALTER TABLESPACE fsdata READ WRITE;
SQL> ALTER TABLESPACE fsindex READ WRITE;
```

    c. Exit from SQL*Plus.
5. On the Database service compute node, prepare the database by creating the required schemas.
    a. On the Database service compute node, invoke SQL*Plus and log in to the PDB as the `SYSTEM` user.
    b. If the owners of the objects that will be imported do not exist in the database, create them before performing the `RESTORE`.

```
SQL> CREATE USER fsowner
  2  PROFILE default
  3  IDENTIFIED BY fspass
  4  TEMPORARY TABLESPACE temp
  5  ACCOUNT UNLOCK;
```

6. On the Database service compute node, invoke RMAN and use the `RESTORE` command with the `foreignFileSpec` subclause to restore the cross-platform backup.
    a. Create an operating system directory for the Data Pump Dump file.

```
mkdir /tmp/from_onprem
```

    b. Invoke RMAN and log in to the PDB as a user that has been granted the `SYSDBA` or `SYSBACKUP` privilege.

```
rman target username@pdb_servicename
```

    c. Execute the `RESTORE` command.

```
RMAN> RESTORE FOREIGN TABLESPACE fsdata,fsindex TO NEW
2> FROM BACKUPSET '/tmp/from_onprem/fs_tbs.bck'
3> DUMP FILE DATAPUMP DESTINATION '/tmp/datapump'
4> FROM BACKUPSET '/tmp/from_onprem/fs_tbs.dmp';
```

d. Exit from RMAN.

7. On the Database service compute node, set the tablespaces to READ WRITE mode.
   a. Invoke SQL*Plus and log in to the PDB as the `SYSTEM` user.
   b. Set the `FSDATA` and `FSINDEX` tablespaces to READ WRITE.

```
SQL> ALTER TABLESPACE fsdata READ WRITE;
SQL> ALTER TABLESPACE fsindex READ WRITE;
```

   c. Exit from SQL*Plus.
8. After verifying that the data has been imported successfully, you can delete the backup set files that were transported from the on-premises host.

RMAN Transportable Tablespace with Data Pump

You can use this method only if the on-premises platform is little endian, and the database character sets of your on-premises database and the Databaseservice database are compatible.

You can use this method to eliminate placing the tablespaces in `READ ONLY` mode, as required by the Data Pump Transportable Tablespace method.

To migrate an on-premises source database to a database deployment on the Database service using the RMAN Transportable Tablespace with Data Pump method, you perform these tasks:

1. On the on-premises database host, invoke RMAN and create the transportable tablespace set.
2. Use a secure copy utility to transfer the Data Pump Export dump file and the tablespace datafiles to the Database service compute node.
3. On the Database service compute node, prepare the database for the tablespace import.

4. On the Database service compute node, invoke Data Pump Import and connect to the database. Import the data into the database using the `TRANSPORT_DATAFILES` option.
5. After verifying that the data has been imported successfully, you can delete the dump file.

RMAN Transportable Tablespace with Data Pump: Example

This example provides a step-by-step demonstration of the tasks required to migrate tablespaces in an on-premises Oracle database to a Database service database.

This example performs a migration of the `FSDATA` and `FSINDEX` tablespaces.

In this example, the on-premises database is on a Linux host.

1. On the on-premises database host, invoke RMAN and create the transportable tablespace set.
    a. On the on-premises database host, create an operating system directory for the datafiles.

    ```
    mkdir /u01/app/oracle/admin/orcl/rman_transdest
    ```

    b. On the on-premises data host, create an operating system directory for the RMAN auxiliary instance files.

    ```
    mkdir /u01/app/oracle/admin/orcl/rman_auxdest
    ```

    c. Invoke RMAN and log in as the `SYSTEM` user. Enter the password for the `SYSTEM` user when prompted.

    ```
    rman target system
    ```

    d. Execute the `TRANSPORT TABLESPACE` command.

    ```
    RMAN> TRANSPORT TABLESPACE fsdata, fsindex
    2> TABLESPACE DESTINATION '/u01/app/oracle/admin/orcl/rman_transdest'
    3> AUXILIARY DESTINATION '/u01/app/oracle/admin/orcl/rman_auxdest';
    ```

    e. Log out of RMAN.
    f. Optionally, navigate to the directory you specified for the `TABLESPACE`

DESTINATION and view the files that were created by the TRANSPORT TABLESPACE operation.

```
cd /u01/app/oracle/admin/orcl/rman_transdest
$ ls
dmpfile.dmp  fsdata01.dbf  fsindex01.dbf  impscrpt.sql
```

2. Use a secure copy utility to transfer the Data Pump Export dump file and the tablespace datafiles to the Database service compute node.

   In this example the dump file is copied to the /u01 directory. Choose the appropriate location based on the size of the file that will be transferred.

   a. On the Database service compute node, create a directory for the dump file.

   ```
   mkdir /u01/app/oracle/admin/ORCL/dpdump/from_onprem
   ```

   b. Before using the scp command to copy files, make sure the SSH private key that provides access to the Database service compute node is available on your on-premises host.

   c. On the on-premises database host, use the SCP utility to transfer the dump file and all datafiles of the transportable set to the Database service compute node.

   ```
   scp -i private_key_file \
   /u01/app/oracle/admin/orcl/rman_transdest/dmpfile.dmp \
   oracle@IP_address_DBaaS_VM:/u01/app/oracle/admin/ORCL/dpdump/from_onprem

   $ scp -i private_key_file \
   /u01/app/oracle/admin/orcl/rman_transdest/fsdata01.dbf \
   oracle@IP_address_DBaaS_VM:/u02/app/oracle/oradata/ORCL

   $ scp -i private_key_file \
   /u01/app/oracle/admin/orcl/rman_transdest/fsindex01.dbf \
   oracle@IP_address_DBaaS_VM:/u02/app/oracle/oradata/ORCL
   ```

3. On the Database service compute node, prepare the database for the tablespace import.

   a. On the Database service compute node, invoke SQL*Plus and log in to the database as the SYSTEM user.

   b. Create a directory object in the Database service database.

```
SQL> CREATE DIRECTORY dp_from_onprem AS '/u01/app/oracle/admin/ORCL/dpdump/from_onprem';
```

c. If the owners of the objects that will be imported do not exist in the database, create them before performing the import. The transportable tablespace mode of import does not create the users.

```
SQL> CREATE USER fsowner
  2  PROFILE default
  3  IDENTIFIED BY fspass
  4  TEMPORARY TABLESPACE temp
  5  ACCOUNT UNLOCK;
```

4. On the Database service compute node, invoke Data Pump Import and connect to the database.

   Import the data into the database using the `TRANSPORT_DATAFILES` option.

```
impdp system DIRECTORY=dp_from_onprem DUMPFILE='dmpfile.dmp' \
TRANSPORT_DATAFILES='/u02/app/oracle/oradata/ORCL/fsdata01.dbf', \
'/u02/app/oracle/oradata/ORCL/fsindex01.dbf'
```

5. After verifying that the data has been imported successfully, you can delete the `dmpfile.dmp` dump file.

RMAN CONVERT Transportable Tablespace with Data Pump

You can use this method only if the database character sets of your on-premises database and the Database service database are compatible.

This method is similar to the Data Pump Transportable Tablespace method, with the addition of the RMAN `CONVERT` command to enable transport between platforms with different endianness. Query `V$TRANSPORTABLE_PLATFORM` to determine if the on-premises database platform supports cross-platform tablespace transport and to determine the endian format of the platform. The Database service platform is little-endian format.

To migrate tablespaces from your on-premises Oracle database to a database deployment on the Database service using RMAN, you perform these tasks:

1. On the on-premises database host, prepare the database for the Data Pump transportable tablespace export.

2. On the on-premises database host, invoke Data Pump Export to perform the transportable tablespace export.
3. On the on-premises database host, invoke RMAN and use the `CONVERT TABLESPACE` command to convert the tablespace datafile to the Database service platform format. Refer to the Oracle Database Backup and Recovery Reference for more information on the `CONVERT` command.
4. Use a secure copy utility to transfer the Data Pump Export dump file and the converted tablespace datafiles to the Database service compute node.
5. Set the on-premises tablespaces back to `READ WRITE`.
6. On the Database service compute node, prepare the database for the tablespace import.

7. On the Database service compute node, invoke Data Pump Import and connect to the database.
8. On the Database service compute node, set the tablespaces in the database to `READ WRITE` mode.
9. After verifying that the data has been imported successfully, you can delete the dump file.

RMAN CONVERT Transportable Tablespace with Data Pump: Example

This example provides a step-by-step demonstration of the tasks required to migrate tablespaces in an on-premises Oracle database to a Database service database.

In this example, the on-premises database is on a Linux host.

1. On the on-premises database host, prepare the database for the Data Pump transportable tablespace export.
    a. On the on-premises database host, create a directory in the operating system to use for the on-premises export.

    ```
    mkdir /u01/app/oracle/admin/orcl/dpdump/for_cloud
    ```

    b. On the on-premises database host, invoke SQL*Plus and log in to the on-premises

database as the SYSTEM user.

```
sqlplus system
Enter password: <enter the password for the SYSTEM user>
```

c. Create a directory object in the on-premises database to reference the operating system directory.

```
SQL> CREATE DIRECTORY dp_for_cloud AS '/u01/app/oracle/admin/orcl/dpdump/for_cloud';
```

d. On the on-premises database host, set all tablespaces that will be transported (the transportable set) to READ ONLY mode.

```
SQL> ALTER TABLESPACE fsindex READ ONLY;
Tablespace altered.
SQL> ALTER TABLESPACE fsdata READ ONLY;
Tablespace altered.
```

e. Exit from SQL*Plus.

2. On the on-premises database host, invoke Data Pump Export to perform the transportable tablespace export.

   On the on-premises database host, invoke Data Pump Export and connect to the on-premises database. Export the on-premises tablespaces using the TRANSPORT_TABLESPACES option. Provide the password for the SYSTEM user when prompted.

```
expdp system TRANSPORT_TABLESPACES=fsdata,fsindex TRANSPORT_FULL_CHECK=YES DIRECTORY=dp_for_cloud
```

3. On the on-premises database host, invoke RMAN and use the CONVERT TABLESPACE command to convert the tablespace datafile to the Database service platform format.

   a. Invoke RMAN.

```
rman target /
```

   b. Execute the RMAN CONVERT TABLESPACE command to convert the datafiles and store the converted files in a temporary location on the on-premises database host.

```
RMAN> CONVERT TABLESPACE fsdata, fsindex
   2> TO PLATFORM 'Linux x86 64-bit'
   3> FORMAT '/tmp/%U ';
…
```

```
input datafile file number=00006 name=/u01/app/oracle/oradata/orcl/fsdata01.dbf
converted datafile=/tmp/data_D-ORCL_I-1410251631_TS-FSDATA_FNO-6_0aqc9un3
…
input datafile file number=00007 name=/u01/app/oracle/oradata/orcl/fsindex01.dbf
converted datafile=/tmp/data_D-ORCL_I-1410251631_TS-FSINDEX_FNO-7_0bqc9un6
…
```

c. Take note of the names of the converted files. You will copy these files to the Database service compute node in the next step.

d. Exit RMAN.

4. Use a secure copy utility to transfer the Data Pump Export dump file and the converted tablespace datafiles to the Database service compute node.

   In this example the dump file is copied to the `/u01` directory. Choose the appropriate location based on the size of the file that will be transferred.

   a. On the Databaseservice compute node, create a directory for the dump file.

   ```
   mkdir /u01/app/oracle/admin/ORCL/dpdump/from_onprem
   ```

   b. Before using the `scp` command to copy files, make sure the SSH private key that provides access to the Database service compute node is available on your on-premises host.

   c. On the on-premises database host, use the `scp` utility to transfer the dump file and all data files of the transportable set to the Database service compute node.

   ```
   scp -i private_key_file \
   /u01/app/oracle/admin/orcl/dpdump/for_cloud/expdat.dmp \
   oracle@IP_address_DBaaS_VM:/u01/app/oracle/admin/ORCL/dpdump/from_onprem

   $ scp -i private_key_file \
   /tmp/data_D-ORCL_I-1410251631_TS-FSDATA_FNO-6_0aqc9un3 \
   oracle@IP_address_DBaaS_VM:/u02/app/oracle/oradata/ORCL/fsdata01.dbf

   $ scp -i private_key_file \
   /tmp/data_D-ORCL_I-1410251631_TS-FSINDEX_FNO-7_0bqc9un6 \
   oracle@IP_address_DBaaS_VM:/u02/app/oracle/oradata/ORCL/fsindex01.dbf
   ```

5. Set the on-premises tablespaces back to `READ WRITE`.

a. Invoke SQL*Plus and log in as the `SYSTEM` user.

b. Set the `FSDATA` and `FSINDEX` tablespaces back to `READ WRITE` mode.

```
SQL> ALTER TABLESPACE fsdata READ WRITE;
Tablespace altered.
SQL> ALTER TABLESPACE fsindex READ WRITE;
Tablespace altered.
```

c. Exit from SQL*Plus.

6. On the Database service compute node, prepare the database for the tablespace import.

a. On the Database service compute node, invoke SQL*Plus and log in to the database as the `SYSTEM` user.

b. Create a directory object in the Database service database.

```
SQL> CREATE DIRECTORY dp_from_onprem AS '/u01/app/oracle/admin/ORCL/dpdump/from_onprem';
```

c. If the owners of the objects that will be imported do not exist in the database, create them before performing the import. The transportable tablespace mode of import does not create the users.

```
SQL> CREATE USER fsowner
  2  PROFILE default
  3  IDENTIFIED BY fspass
  4  TEMPORARY TABLESPACE temp
  5  ACCOUNT UNLOCK;
```

7. On the Database service compute node, invoke Data Pump Import and connect to the database.

   Import the data into the Database service database using the `TRANSPORT_DATAFILES` option

```
impdp system DIRECTORY=dp_from_onprem \
TRANSPORT_DATAFILES='/u02/app/oracle/oradata/ORCL/fsdata01.dbf', \
'/u02/app/oracle/oradata/ORCL/fsindex01.dbf'
```

8. On the Database service compute node, set the tablespaces in the database to `READ WRITE` mode.

a. Invoke SQL*Plus and log in as the `SYSTEM` user.

b. Set the `FSDATA` and `FSINDEX` tablespaces to `READ WRITE` mode.

```
SQL> ALTER TABLESPACE fsdata READ WRITE;
Tablespace altered.
SQL> ALTER TABLESPACE fsindex READ WRITE;
Tablespace altered.
```

c. Exit from SQL*Plus.

9. After verifying that the data has been imported successfully, you can delete the `expdat.dmp` dump file.

RMAN DUPLICATE from an Active Database

This topic explains how to migrate an entire, active container database (CDB) or non-CDB database to Oracle Cloud Infrastructure by using RMAN Active Duplication. The database to be migrated can reside on-premises or in Oracle Cloud Infrastructure Classic. This topic does not cover duplicating a pluggable database, or migrating a pluggable database or non-CDB to a CDB in the cloud.

The following terms are used throughout this topic:

- Source database: The active database to be migrated.
- Target database: The new database (duplicated from the source database) on a DB system in the Oracle Cloud Infrastructure.



Note

Version 11.2.0.4 databases will be migrated to a DB system using a ACFS storage.

Prerequisites

For the source database to be migrated, you'll need:

- The source database name, database unique name, listener port, service name, database home patch level, and the password for SYS.
- A copy of the sqlpatch directory from the source database home. This is required for rollback in case the target DB system does not include these patches.
- If the source database is configured with Transparent Data Encryption (TDE), you'll need a backup of the wallet and the wallet password to allow duplication of a database with encrypted data.

When migrating a source database to an existing target database, Oracle recommends that you patch the source environment to the same database bundle patch level as the target database home. If the source environment has an interim patch (previously known as a "one-off" patch) that includes a sqlpatch component, and that sqlpatch is missing from the target environment (or a different cumulative patch is applied), the interim patch should be rolled back in the source environment before the migration, if possible.



Tip

To check for interim patches installed on the source or target database, use the `$ORACLE_HOME/OPatch/opatch lspatches` command. To roll back SQL changes in the target database, copy the `$ORACLE_HOME/sqlpatch/<patch#>/postdeinstall.sql` script from the source environment to the cloud environment and execute the `postdeinstall.sql` script.

For the target database, you'll need:

- A target DB system that supports the same database edition as the source database edition. When you launch a DB system, an initial database is created on it. If necessary, you can delete that database and create a new one by using the dbcli command line interface. For more information on creating a DB system, see Managing Bare Metal and Virtual Machine DB Systems. For information about creating a database with the DBCLI,

see Database Commands.

- The target database name, database unique name, auxiliary service name, and database home patch level.
- A free TCP port in the target database to setup the auxiliary instance.

If you need to roll back interim patches in the target environment so that the patch level matches that of the source environment, copy the source DB $ORACLE_HOME/sqlpatch/*<patch_number>* directory to the target database home.

Migrating Source Databases That Include Patch Set Updates (PSUs)

In Oracle Cloud Infrastructure DB systems, the database home includes an installation of Database Proactive Bundle Patches. If the source DB uses Patch Set Updates (PSUs), follow the instructions in MOS Note:1962125.1 (Oracle Database - Overview of Database Patch Delivery Methods) for migrating the DB into Oracle Cloud Infrastructure.

Verifying the Environment

Perform the following steps before you begin the migration:

1. Make sure the source DB system is reachable from the target DB system. You should be able to SSH between the two hosts.
2. On the target host, use the TNSPING utility to make sure the source host listener port works. For example:

   ```
   tnsping <source_host>:1521
   ```

3. On the target host, use Easy Connect to verify the connection to the source database:

   ```
   <host>:<port>/<service_name>
   ```

   For example:

   ```
   sqlplus system@129.145.0.164:1521/proddb
   ```

   Make sure the connection string does not exceed 64 characters.
4. Copy the required sqlpatch files (for rollback) from the source database home to the target database.

5. Make sure at least one archivelog has been created on the source database, otherwise, the RMAN duplication will fail with an error.
6. If the source database uses wallets, back up the password-based wallet and copy it to the standard location in the DB system:

```
/opt/oracle/dcs/commonstore/wallets/tde/<db_unique_name>/
```

7. Make sure the compatibility parameters in the source database are set to at least:
   - 18.0.0.0.0 for an 18.1.0.0 database
   - 12.1.0.2.0 for a 12.1.0.2 or a 12.2.0.1 database
   - 11.2.0.4.0 for an 11.2.0.4 database

Setting Up Storage on the DB System

1. SSH to the DB System.

```
ssh -i <private_key_path> opc@<db_system_ip_address>
```

2. Log in as opc and then sudo to the root user. Use `sudo su -` with a hyphen to invoke the root user's profile, which will set the PATH to the dbcli directory (`/opt/oracle/dcs/bin`).

```
login as: opc

[opc@dbsys ~]$ sudo su -
```

3. Use the dbcli create-dbstorage to set up directories for DATA, RECO, and REDO storage. The following example creates 10GB of ACFS storage for the tdetest database.

```
[root@dbsys ~]# dbcli create-dbstorage --dbname tdetest --dataSize 10 --dbstorage ACFS
```



Note

When migrating a version 11.2 database, ACFS storage must be specified.

4. Use the dbcli list-dbstorages command to list the storage ID. You'll need the ID for the

next step.

```
[root@dbsys ~]# dbcli list-dbstorages
ID                                       Type   DBUnique Name        Status
---------------------------------------- ------ -------------------- ----------
9dcdfb8e-e589-4d5f-861a-e5ba981616ed     Acfs   tdetest              Configured
```

5. Use the dbcli describe-dbstorage command with the storage ID from the previous step to list the DATA, RECO and REDO locations.

```
[root@dbsys ~]# dbcli describe-dbstorage --id 9dcdfb8e-e589-4d5f-861a-e5ba981616ed
DBStorage details
----------------------------------------------------------------
                     ID: 9dcdfb8e-e589-4d5f-861a-e5ba981616ed
                DB Name: tdetest
          DBUnique Name: tdetest
         DB Resource ID:
           Storage Type: Acfs
          DATA Location: /u02/app/oracle/oradata/tdetest
          RECO Location: /u03/app/oracle/fast_recovery_area/
          REDO Location: /u03/app/oracle/redo/
                  State: ResourceState(status=Configured)
                Created: August 24, 2016 5:25:38 PM UTC
            UpdatedTime: August 24, 2016 5:25:53 PM UTC
```

Note the locations. You'll use them later to set the `db_create_file_dest`, `db_create_online_log_dest`, and `db_recovery_file_dest` parameters for the database.

Choosing an ORACLE_HOME

Decide which ORACLE_HOME to use for the database restore and then switch to that home with the correct ORACLE_BASE, ORACLE_HOME, and PATH settings.

To get a list of existing ORACLE_HOMEs, use the dbcli list-dbhomes command. To create a new ORACLE_HOME, use the dbcli create-dbhome command.

Copying the Source Database Wallets

Skip this section if the source database is **not** configured with TDE.

1. On the DB system, become the oracle user:

```
sudo su - oracle
```

2. Create the following directory if it does not already exist:

```
mkdir /opt/oracle/dcs/commonstore/wallets/tde/<db_unique_name>
```

3. Copy the ewallet.p12 file from the source database to the directory you created in the previous step.
4. On the target host, make sure that `$ORACLE_HOME/network/admin/sqlnet.ora` contains the following line:

```
ENCRYPTION_WALLET_LOCATION=(SOURCE=(METHOD=FILE)(METHOD_DATA=
(DIRECTORY=/opt/oracle/dcs/commonstore/wallets/tde/$ORACLE_UNQNAME)))
```

   Add the line if it doesn't exist in the file. (The line might not be there if this is a new home and no database has been created yet on this host.)
5. Create the autologin wallet from the password-based wallet to allow auto-open of the wallet during restore and recovery operations.

   For version 12c, use the `ADMINISTER KEY MANAGEMENT` command:

```
$cat create_autologin_12.sh

#!/bin/sh
if [ $# -lt 2 ]; then
        echo "Usage: $0 <db_unique_name> <remote_wallet_location>"
        exit 1;
fi

mkdir /opt/oracle/dcs/commonstore/wallets/tde/$1
cp $2/ewallet.p12* /opt/oracle/dcs/commonstore/wallets/tde/$1
rm -f autokey.ora
echo "db_name=$1"  > autokey.ora
autokeystoreLog="autologinKeystore_`date +%Y%m%d_%H%M%S_%N`.log"
echo "Enter Keystore Password:"
read -s keystorePassword
echo "Creating AutoLoginKeystore -> "
sqlplus "/as sysdba"  <<EOF
spool $autokeystoreLog
set echo on
```

```
startup nomount pfile=autokey.ora
ADMINISTER KEY MANAGEMENT CREATE AUTO_LOGIN KEYSTORE
FROM KEYSTORE '/opt/oracle/dcs/commonstore/wallets/tde/$1' -- Keystore location
IDENTIFIED BY "$keystorePassword";
shutdown immediate;
EOF
```

For version 11g, use the `orapki` command:

```
orapki wallet create -wallet wallet_location -auto_login [-pwd <password>]
```

Setting Up the Static Listener

Set up the static listener for the auxiliary instance for RMAN duplication.

1. On the DB system, create `$ORACLE_HOME/network/admin/listener.ora` and add the following content to it.

```
LISTENER_aux_<db_unique_name>=
  (DESCRIPTION=
    (ADDRESS_LIST=
      (ADDRESS=(PROTOCOL=TCP)(HOST=<hostname> or <ip_address>)(PORT=<available_TCP_port>))
    )
   )
SID_LIST_LISTENER_aux_<db_unique_name>=
  (SID_LIST=
    (SID_DESC=
      (GLOBAL_DBNAME=<auxServiceName_with_domain>)
       (ORACLE_HOME=<Oracle_home_for_target_database>)
       (SID_NAME=<database_name>)
       (ENVS="TNS_ADMIN=<path_to_tnsnames.ora>")
       (ENVS="ORACLE_UNQNAME=<db_unique_name(in lower case)>"))
)
```

2. Make sure the port specified in `(PORT=<available_TCP_port>)` is open in the DB system's iptables and in the DB system's cloud network Security List.

Using the RMAN Duplicate Command to Migrate the Database

1. Set the following environment variables for RMAN and SQL Plus sessions for the database:

```
ORACLE_HOME=<path_of_Oracle_home_where_the database_is_to_be_restored>
ORACLE_SID=<database_name>
ORACLE_UNQNAME=<db_unique_name(in lower case)>
NLS_DATE_FORMAT="mm/dd/yyyy hh24:mi:ss"
```

2. Start the listener:

```
lsnrctl start listener_aux_<db_unique_name>
```

3. Create an `init.ora` file with the minimal required parameters as described in Creating an Initialization Parameter File and Starting the Auxiliary Instance and use it for the auxiliary instance.
4. Start the auxiliary instance in nomount mode:

```
startup nomount
```

5. Run the following commands to duplicate the database. Note that the example below uses variables to indicate the values to be specified:

```
rman target sys/$sourceSysPassword@$sourceNode:$sourceListenerPort/$sourceDb auxiliary
sys/$auxSysPassword@$targetNode:$targetListenerPort/$auxService<<EOF

spool log to "`date +%Y%m%d_%H%M%S_%N`_duplicate_${targetDbUniqueName}_from_${sourceDb}.log"
set echo on

duplicate target database to $targetDb from active database
password file
spfile
 PARAMETER_VALUE_CONVERT $sourceDb $targetDb $sourceDbUniqueNameCaps $targetDbUniqueNameCaps
 set cluster_database='false'
 set db_name='$targetDb'
 set db_unique_name='$targetDbUniqueName'
 set db_create_file_dest='$dataLoc'
 set db_create_online_log_dest_1='$redoLoc'
 set db_recovery_file_dest='$recoLoc'
 set audit_file_dest = '$auditFileDest'
 reset control_files
nofilenamecheck
;
EOF
```

Preparing to Register the Database

Before you register the database:

1. Make sure the database COMPATIBLE parameter value is acceptable.
   For a 11.2 database, the minimum compatibility value is 11.2.0.4.
   For a 12c database, the minimum compatibility value is 12.1.0.2.
   If the value is less than the minimum, the database cannot be registered until you upgrade the database compatibility.
2. Use the following command to verify that the database has registered with the local listener and service name.

   ```
   lsnrctl services
   ```

3. Use the following command to verify that the password file was restored or created for a new database.

   ```
   ls -ltr $ORACLE_HOME/dbs/orapw<$ORACLE_SID>
   ```

   If the file does not exist, create it using the orapwd command.

   ```
   orapwd file=<$ORACLE_HOME/dbs/orapw<$ORACLE_SID>> password=<sys_password>
   ```

4. Use the following command to verify that the restored database is open in read write mode.

   ```
   select open_mode from v$database;
   ```

   Read write mode is required to register the database later. Any PDBs must also be in read write mode.
5. From oracle home on the migrated database host, use the following command verify the connection to SYS.

   ```
   conn sys/<password>@<service_name> as sysdba
   ```

   This connection is required to register the database later. Fix any connection issues before continuing.
6. Copy the folder `$ORACLE_HOME/sqlpatch` from source database to the target database. This will enable the `dbcli register-database` command to rollback any conflicting patches.



Note

If you are migrating a version 11.2 database, additional steps are required after you register the database. For more information, see Rolling Back Patches on a Version 11.2 Database.

7. Use the following SQL*Plus command to make sure the database is using the spfile.

```
SHOW PARAMETERS SPFILE
```

Registering the Database on the DB System

The dbcli register-database command registers the migrated database to the dcs-agent so it can be managed by the dcs-agent stack.



Note

The `dbcli register-database` command is not available on 2-node RAC DB Systems.

As the root user, use the `dbcli register-database` command to register the database on the DB system, for example:

```
[root@dbsys ~]# dbcli register-database --dbclass OLTP --dbshape odb1 --servicename crmdb.example.com --
syspassword
Password for SYS:
{
  "jobId" : "317b430f-ad5f-42ae-bb07-13f053d266e2",
  "status" : "Created",
  "message" : null,
  "reports" : [ ],
  "createTimestamp" : "August 08, 2016 05:55:49 AM EDT",
  "description" : "Database service registration with db service name: crmdb.example.com",
  "updatedTime" : "August 08, 2016 05:55:49 AM EDT"
}
```

Migrating a Version 12.1 or Later Database That Includes SQL Patch Components

For a 1-node DB system at version 12.1 or higher, the `dbcli register-database` command automates the datapatch execution. Before executing the `dbcli register-database` command, open all PDBs in read-write mode. If you have already run the `dbcli register-database` command and did not open all PDBs, or did not copy the $ORACLE_HOME/sqlpatch directory from the source database home, manually rerun the datapatch utility to configure the SQL portion of existing interim patches. This can be done by executing the command `$ORACLE_HOME/OPatch/opatch datapatch`.



Tip

If the source database includes patch 23170620 and the target database is running with the October 2017 patch or a later one, the $ORACLE_HOME/sqlpatch directory does not need to be copied to the target database, because the contents of the patch are already installed in the target database.

Rolling Back Patches on a Version 11.2 Database

For version 11.2 databases, the sqlpatch application is not automated, so any interim patches (previously known as a "one-off" patches) applied to the source database that are not part of the installed PSU must be rolled back manually in the target database. After registering the database, execute the `catbundle.sql` script and then the `postinstall.sql` script with the corresponding PSU patch (or the overlay patch on top of the PSU patch), as described below.



Tip

Some interim patches may include files written to the $ORACLE_HOME/rdbms/admin directory as well as the $ORACLE_HOME/sqlpatch directory. Oracle recommends that you roll back these patches in the source database using the instructions in the patch read-me prior to migrating the database to OCI environment. Contact Oracle Support if you need assistance with rolling back these patches.

1. On the DB System, use the `dbcli list-dbhomes` command to find the PSU patch number for the version 11.2 database home. In the following sample command output, the PSU patch number is the second number in the DB Version column:

```
[root@dbsys ~]# dbcli  list-dbhomes
ID                                   Name                DB Version
Home Location                                Status
------------------------------------ ------------------  ------------------------------------  --
---------------------------------------- ----------
59d9bc6f-3880-4d4f-b5a6-c140f16f8c64 OraDB11204_home1  11.2.0.4.160719 (23054319, 23054359)
/u01/app/oracle/product/11.2.0.4/dbhome_1 Configured
```

   (The first patch number, 23054319 in the example above, is for the OCW component in the database home.)

2. Find the overlay patch, if any, by using the `lsinventory` command. In the following example, patch number *24460960* is the overlay patch on top of the 23054359 PSU patch.

```
$ $ORACLE_HOME/OPatch/opatch lsinventory
...
Installed Top-level Products (1):

Oracle Database 11g                                                  11.2.0.4.0
There are 1 products installed in this Oracle Home.
```

```
Interim patches (5) :

Patch  24460960     : applied on Fri Sep 02 15:28:17 UTC 2016
Unique Patch ID:  20539912
   Created on 31 Aug 2016, 02:46:31 hrs PST8PDT
   Bugs fixed:
     23513711, 23065323, 21281607, 24006821, 23315889, 22551446, 21174504
   This patch overlays patches:
     23054359
   This patch needs patches:
     23054359
   as prerequisites
```

3. Start SQL*Plus and execute the `catbundle.sql` script, for example:

```
SQL> startup
SQL> connect / as sysdba
SQL> @$ORACLE_HOME/rdbms/admin/catbundle.sql psu apply
exit
```

4. Apply the sqlpatch, using the overlay patch number from the previous step, for example:

```
SQL> connect / as sysdba
SQL> @$ORACLE_HOME/sqlpatch/24460960/postinstall.sql
exit
```

Creating a Backup Configuration (Optional)

If you would like to manage the database backup with the dbcli command line interface, you can associate a new or existing backup configuration with the migrated database when you register it or after you register it. A backup configuration defines the backup destination and recovery window for the database. As the root user, use the following commands to create, list, and display backup configurations:

- dbcli update-backupconfig
- dbcli list-backupconfigs
- dbcli describe-backupconfig

Post Migration Checklist

After the database is migrated and registered on the DB system, use the following checklist to verify the results of the migration and perform any post-migration customizations.

1. Make sure the database files were restored in OMF format.
2. Make sure the database is listed in the dbcli list-databases command output.
3. Check for the following external references in the database and update them if necessary:
    - External tables: If the source database uses external tables, back up that data and migrate it to the target host.
    - Directories: Customize the default directories as needed for the migrated database.
    - Database links: Make sure all the required TNS entries are updated in the tnsnames.ora file in ORACLE_HOME.
    - Email and URLs: Make sure any email addresses and URLs used in the database are still accessible from the DB system.
    - Scheduled jobs: Review the jobs scheduled in source database and schedule similar jobs as needed in the migrated database.
4. If you associated a backup configuration when you registered the database, run a test back up using the dbcli create-backup command.
5. Verify that patches have been applied to all PDBs if the migrated database contains CDB and PDBs.
6. Validate the database performance by using Database Replay and SQL Performance Analyzer for SQL. For more information, see the Database Testing Guide.

SQL Developer and INSERT Statements to Migrate Selected Objects

You can use SQL Developer to create a cart into which you add selected objects to be loaded into an Oracle Database 12c database in the Oracle Cloud Infrastructure Database service.

In this method, you use SQL `INSERT` statements to load the data into your cloud database.

To migrate selected objects to an Oracle Database 12c database in a Database service deployment using SQL Developer and `INSERT` statements, you perform these tasks:

1. Launch SQL Developer, connect to your on-premises database and create a cart containing the objects you want to migrate.
2. In SQL Developer, click the Export Cart icon and select "Insert" in the Format menu.
3. In SQL Developer, open a connection to the Oracle Database 12c database in the Database service and execute the generated script to create the database objects.
4. In SQL Developer, open a connection to the Oracle Database 12c database in the Database service and run the generated script to create the objects and load the data.

SQL Developer and SQL*Loader to Migrate Selected Objects

You can use SQL Developer to create a cart into which you add selected objects to be loaded into an Oracle Database 12c database in the Oracle Cloud Infrastructure Database.

In this method, you use SQL*Loader to load the data into your cloud database.

To migrate selected objects to an Oracle Database 12c database in the Database service deployment using SQL Developer and SQL*Loader, you perform these tasks:

1. Launch SQL Developer, connect to your on-premises database and create a cart containing the objects you want to load into your cloud database.
2. In SQL Developer, click the Export Cart icon and select "loader" in the Format menu.
3. In SQL Developer, open a connection to the Oracle Database 12c database on the Database service and execute the generated script to create the database objects.
4. Use a secure copy utility to transfer the SQL*Loader control files and the SQL*Loader data files to the the Database service compute node.
5. On the the Database service compute node, invoke SQL*Loader to load the data using the SQL*Loader control files and data files for each object.

Unplugging/Plugging a PDB

You can use this method only if the on-premises platform is little endian, and the on-premises database and the Oracle Cloud Infrastructure Database service database have compatible database character sets and national character sets.

You can use the unplug/plug method to migrate an Oracle Database 12c PDB to a PDB in an Oracle Database 12c database on a Database service database deployment.

To migrate an Oracle Database 12c PDB to a PDB in the Oracle Database 12c database on an Oracle Cloud Infrastructure Database service database deployment using the plug/unplug method, you perform these tasks:

1. On the on-premises database host, invoke SQL*Plus and close the on-premises PDB.
2. On the on-premises database host, execute the `ALTER PLUGGABLE DATABASE UNPLUG` command to generate an XML file containing the list of datafiles that will be plugged in to the database on the Database service.
3. Use a secure copy utility to transfer the XML file and the datafiles to the Databaseservice compute node.
4. On the Database service compute node, invoke SQL*Plus and execute the `CREATE PLUGGABLE DATABASE` command to plug the database into the CDB.
5. On the Database service compute node, open the new PDB by executing the `ALTER PLUGGABLE DATABASE OPEN` command.

For more information, see "Creating a PDB by Plugging an Unplugged PDB into a CDB" in *Oracle Database Administrator's Guide* for Release 12.2 or 12.1.

Unplugging/Plugging Non-CDB

You can use this method only if the on-premises platform is little endian, and the on-premises database and the Oracle Cloud Infrastructure Database database have compatible database character sets and national character sets.

You can use the unplug/plug method to migrate an Oracle Database 12c non-CDB database to a PDB in an Oracle Database 12c database on a Database service database deployment. This

method provides a way to consolidate several non-CDB databases into a single Oracle Database 12c multitenant database on the Database service.

To migrate an Oracle Database 12c non-CDB database to the Oracle Database 12c database on a Database service deployment using the plug/unplug method, you perform these tasks:

1. On the on-premises database host, invoke SQL*Plus and set the on-premises database to `READ ONLY` mode.
2. On the on-premises database host, execute the `DBMS_PDB.DESCRIBE` procedure to generate an XML file containing the list of datafiles that will be plugged in on the cloud database.
3. Use a secure copy utility to transfer the XML file and the datafiles to the Database service compute node.
4. On the Database service compute node, invoke SQL*Plus and execute the `CREATE PLUGGABLE DATABASE` command to plug the database into the CDB.
5. On the Database service compute node, execute the `$ORACLE_HOME/rdbms/admin/noncdb_to_pdb.sql` script to delete unnecessary metadata from the `SYSTEM` tablespace of the new PDB.
6. On the Database service compute node, open the new PDB by executing the `ALTER PLUGGABLE DATABASE OPEN` command.
7. Optionally, on the on-premises database host invoke SQL*Plus and set the on-premises database back to `READ WRITE` mode.

For more information, see "Creating a PDB Using a Non-CDB" in *Oracle Database Administrator's Guide* for Release 12.2 or 12.1.

Troubleshooting

These topics cover some common issues you might run into and how to address them.

- Backup Failures
- Patching Failures

Backup Failures

Database backups can fail for various reasons. Typically, a backup fails because either the database host cannot access the object store, or there are problems on the host or with the database configuration.

This topic includes information to help you determine the cause of the failure and fix the problem. The information is organized into several sections, based on the error condition. If you already know the cause, you can skip to the section with the suggested solution. Otherwise, use the procedure in Determining the Problem to get started.

Determining the Problem

In the Console, a failed database backup either displays a status of **Failed** or hangs in the **Backup in Progress** or **Creating** state. If the error message does not contain enough information to point you to a solution, you can use the database CLI and log files to gather more data. Then, refer to the applicable section in this topic for a solution.

To identify the root cause of the backup failure

1. Log on to the host as the root user and navigate to the `/opt/oracle/dcs/bin/` directory.
2. Determine the sequence of operations performed on the database.

   ```
   dbcli list-jobs | grep -i <dbname>
   ```

   Note the last job ID listed with a status other than **Success**.
3. With the job ID you noted from the previous step, use the following command to check the details of that job:

   ```
   dbcli describe-job -i <job_ID> -j
   ```

   Typically, running this command is enough to reveal the root cause of the failure.
4. If you require more information, review the `/opt/oracle/dcs/log/dcs-agent.log` file.

You can find the job ID in this file by using the timestamp returned by the job report in step 2.

5. If the problem details suggest an RMAN issue, review the RMAN logs in the `/opt/oracle/dcs/log/<hostname>/rman/bkup/<db_unique_name>/rman_backup/<yyyy-mm-dd>` directory.



Note

If the database failure is on a 2-node RAC database, perform steps 3 and 4 on both nodes.

Database Service Agent Issues

Your Oracle Cloud Infrastructure Database makes use of an agent framework to allow you to manage your database through the cloud platform. Occasionally you might need to restart the dcsagent program if it has the status of **stop/waiting** to resolve a backup failure.

To restart the database service agent

1. From a command prompt, check the status of the agent:

```
initctl status initdcsagent
```

2. If the agent is in the **stop/waiting** state, try to restart the agent:

```
initctl start initdcsagent
```

3. Check the status of the agent again to confirm that it has the **start/running** status:

```
initctl status initdcsagent
```

Oracle Clusterware Issues

Oracle Clusterware enables servers to communicate with each other so that they can function as a collective unit. Occasionally you might need to restart the Clusterware program to

resolve a backup failure.

To restart the Oracle Clusterware

1. From command prompt, check the status of Oracle Clusterware:

   ```
   crsctl check crs
   ```

   ```
   crsctl stat res -t
   ```

2. If Oracle Clusterware is not online, try to restart the program:

   ```
   crsctl start crs
   ```

3. Check the status of Oracle Clusterware to confirm that it is online:

   ```
   crsctl check crs
   ```

Object Store Connectivity Issues

Backing up your database to Oracle Cloud Infrastructure Object Storage requires that the host can connect to the applicable Swift endpoint. You can test this connectivity by using a Swift user.

To ensure your database host can connect to the object store

1. Create a Swift user in your tenancy. See Working with Auth Tokens.
2. With the user you created in the previous step, use the following command to verify the host can access the object store.

   ```
   curl -v -X HEAD -u <user_ID>:'<auth_token>' https://swiftobjectstorage.<region_
   name>.oraclecloud.com/v1/<tenant>
   ```

   See https://cloud.oracle.com/infrastructure/storage/object-storage/faq for the correct region to use.
3. If you cannot connect to the object store, refer to Prerequisites for how to configure object store connectivity.

Host Issues

One or more of the following conditions on the database host can cause backups to fail:

Interactive Commands in the Oracle Profile

If an interactive command such as `oraenv`, or any command that might return an error or warning message, was added to the `.bash_profile` file for the grid or oracle user, Database service operations like automatic backups can be interrupted and fail to complete. Check the `.bash_profile` file for these commands, and remove them.

The File System Is Full

Backup operations require space in the `/u01` directory on the host file system. Use the `df -h` command on the host to check the space available for backups. If the file system has insufficient space, you can remove old log or trace files to free up space.

Incorrect Version of the Oracle Database Cloud Backup Module

Your system might not have the required version of the backup module (opc_installer.jar). See Unable to use Managed Backups in your DB System for details about this known issue. To fix the problem, you can follow the procedure in that section or simply update your DB system and database with the latest bundle patch.

Changes to the Site Profile File (glogin.sql)

Customizing the site profile file (`$ORACLE_HOME/sqlplus/admin/glogin.sql`) can cause managed backups to fail in Oracle Cloud Infrastructure. In particular, interactive commands can lead to backup failures. Oracle recommends that you not modify this file for databases hosted in Oracle Cloud Infrastructure.

Database Issues

An improper database state or configuration can lead to failed backups.

Database Not Running During Backup

The database must be active and running while the backup is in progress.

To check that the database is active and running

Use the following command to check the state of your database, and ensure that any problems that might have put the database in an improper state are resolved:

```
srvctl status database -d <db_unique_name> -verbose
```

The system returns a message including the database's instance status. The instance status must be **Open** for the backup to succeed. If the database is not running, use the following command to start it:

```
srvctl start database -d <db_unique_name> -o open
```

If the database is mounted but does not have the **Open** status, use the following commands to access the SQL*Plus command prompt and set the status to **Open**:

```
sqlplus / as sysdba
```

```
alter database open;
```

ARCHIVING MODE SET TO NOARCHIVELOG

When you provision a new database, the archiving mode is set to `ARCHIVELOG` by default. This is the required archiving mode for backup operations. Check the archiving mode setting for the database and change it to `ARCHIVELOG`, if applicable.

To check and set the archiving mode

Open an SQL*Plus command prompt and enter the following command:

```
select log_mode from v$database;
```

If you need to set the archiving mode to `ARCHIVELOG`, start the database in **Mount** status (and not **Open** status), and use the following command at the SQL*Plus command prompt:

```
alter database archivelog;
```

Be sure to confirm that the `db_recovery_file_dest` parameter points to `+RECO`, and that the `log_archive_dest_1` parameter is set to `USE_DB_RECOVERY_FILE_DEST`.

For RAC databases, one instance must have the **Mount** status when enabling archivelog mode. To enable archivelog mode for a RAC database, perform the following steps:

1. Shutdown all database instances:

```
srvctl stop database -d
```

2. Start one of the database instances in mount state:

```
srvctl start instance -d <db_unique_name> -i <instance_name> -o mount
```

3. Access the SQL*Plus command prompt:

```
sqlplus / as sysdba
```

4. Enable archive log mode:

```
alter database archivelog;
```

```
exit;
```

5. Stop the database:

```
srvctl stop instance -d <db_unique_name> -i <instance_name>
```

6. Re-start all database instances:

```
srvctl start database -d <db_unqiue_name>
```

7. At the SQL*Plus command prompt, confirm the archiving mode is set to `ARCHIVELOG`:

```
select log_mode from v$database;
```

Stuck Database Archiver Process and Backup Failures

Backups can fail when the database instance has a stuck archiver process. For example, this can happen when the flash recovery area (FRA) is full. You can check for this condition using the `srvctl status database -db <db_unique_name> -v` command. If the command returns the following output, you must resolve the stuck archiver process issue before backups will succeed:

```
Instance <instance_identifier> is running on node *<node_identifier>. Instance status: Stuck Archiver
```

Refer to ORA-00257:Archiver Error (Doc ID 2014425.1) for information on resolving a stuck archiver process.

After resolving the stuck process, the command should return the following output :

```
Instance <instance_identifier> is running on node *<node_identifier>. Instance status: Open
```

If the instance status does not change after you resolve the underlying issue with the device or resource being full or unavailable, try one of the following workarounds:

- Restart the database using the `srvctl` command to update the status of the database in the clusterware
- Upgrade the database to the latest patchset levels

TEMPORARY TABLESPACE ERRORS

If fixed table statistics are not up to date on the database, backups can fail with errors referencing temporary tablespace present in the `dcs-agent.log` file. For example:

```
select status from v$rman_status where COMMAND_ID=<backup_id>

ERROR at line 1:
ORA-01652: unable to extend temp segment by 128 in tablespace TEMP
```

Gather your fixed table statics as follows to resolve this issue:

```
conn / as sysdba

exec dbms_stats.gather_fixed_objects_stats();
```

RMAN CONFIGURATION AND BACKUP FAILURES

Editing certain RMAN configuration parameters can lead to backup failures in Oracle Cloud Infrastructure. To check your RMAN configuration, use the `show all` command at the RMAN command line prompt.

See the following list of parameters for details about RMAN the configuration settings that should not be altered for databases in Oracle Cloud Infrastructure.

RMAN configuration settings that should not be altered

```
CONFIGURE RETENTION POLICY TO RECOVERY WINDOW OF 30 DAYS;

CONFIGURE CONTROLFILE AUTOBACKUP ON;

CONFIGURE DEVICE TYPE 'SBT_TAPE' PARALLELISM 5 BACKUP TYPE TO COMPRESSED BACKUPSET;

CONFIGURE CHANNEL DEVICE TYPE DISK MAXPIECESIZE 2 G;

CONFIGURE CHANNEL DEVICE TYPE 'SBT_TAPE' MAXPIECESIZE 2 G FORMAT   '%d_%I_%U_%T_%t' PARMS  'SBT_
LIBRARY=/opt/oracle/dcs/commonstore/pkgrepos/oss/odbcs/libopc.so ENV=(OPC_
PFILE=/opt/oracle/dcs/commonstore/objectstore/opc_pfile/1578318329/opc_tiger_iad3c8.ora)';

CONFIGURE ARCHIVELOG DELETION POLICY TO BACKED UP 1 TIMES TO 'SBT_TAPE';

CONFIGURE CHANNEL DEVICE TYPE DISK MAXPIECESIZE 2 G;

CONFIGURE ENCRYPTION FOR DATABASE ON;
```

RMAN Retention Policy and Backup Failures

The RMAN retention policy configuration can be the source of backup failures. Using the REDUNDANCY retention policy configuration instead of the RECOVERY WINDOW policy can lead to backup failures. Be sure to use the RECOVERY WINDOW OF 30 DAYS configuration.

To configure the RMAN retention policy setting

1. Find the database ID using the following command:

   ```
   dbcli list-databases
   ```

2. Find the `BackupConfigId` value for the database using the following command:

   ```
   dbcli describe-database -i <database_id>
   ```

3. Update the retention policy configuration to `RECOVERY WINDOW OF 30 DAYS`:

   ```
   dbcli update-backupconfig -i <backup_config_id> --recoverywindow 30
   ```

Loss of Objectstore Wallet File and Backup Failures

RMAN backups fail when an objectstore wallet file is lost. The wallet file is necessary to enable connectivity to the object store.

To confirm that the objectstore wallet file exists and has the correct permissions

1. Find the database ID using the following command:

   ```
   dbcli list-databases
   ```

2. Find the `BackupConfigId` value for the database using the following command:

   ```
   dbcli describe-database -i <database_id>
   ```

3. Find the `BackupLocation` value for the database using the following command:

   ```
   dbcli describe-backupconfig <backup_config_id>
   ```

4. Find the file path of the backup config parameter file (`opc_<backup_location_value>_BC.ora`) using the following command:

   ```
   locate opc_<backup_location_value>_BC.ora
   ```

   For example:

   ```
   [root@orcl 13aef284-9d6b-4eb6-8751-2988aexample]# locate opc_b9naijWMAXzi9example_BC.ora

   /opt/oracle/dcs/commonstore/objectstore/opc_pfile/13aef284-9d6b-4eb6-8751-2988a9example/opc_
   b9naijWMAXzi9example_BC.ora
   ```

5. Find the file path to the wallet file in the backup config parameter file by inspecting the value stored in the `OPC_WALLET` parameter. To do this, navigate to the directory containing the backup config parameter file and use the following `cat` command:

   ```
   cat <backup_config_parameter_file>
   ```

   For example:

```
[root@orcl 13aef284-9d6b-4eb6-8751-2988aexample]# cat opc_b9naijWMAXzi9example_BC.ora
OPC_HOST=https://swiftobjectstorage.us-ashburn-1.oraclecloud.com/v1/dbbackupiad
OPC_WALLET='LOCATION=file:/opt/oracle/dcs/commonstore/objectstore/wallets/13aef284-9d6b-4eb6-
8751-2988aexample CREDENTIAL_ALIAS=alias_opc'
OPC_CONTAINER=b9naijWMAXzi9example
```

6. Confirm that the `cwallet.sso` file exists in the directory specified in the `OPC_WALLET` parameter, and confirm that the file has the correct permissions. The file permissions should have the octal value of "600" (`-rw-------`). Use the following command:

   ```
   ls -ltr /opt/oracle/dcs/commonstore/objectstore/wallets/<backup_config_id>
   ```

   For example:

   ```
   [root@orcl 13aef284-9d6b-4eb6-8751-2988aexample]# ls -ltr
   /opt/oracle/dcs/commonstore/objectstore/wallets/13aef284-9d6b-4eb6-8751-2988aexample

   total 4

   -rw------- 1 oracle oinstall    0 Apr 20 06:45 cwallet.sso.lck

   -rw------- 1 oracle oinstall 1941 Apr 20 06:45 cwallet.sso
   ```

TDE Wallet and Backup Failures

Incorrect TDE Wallet Location Specification

For backup operations to work, the `$ORACLE_HOME/network/admin/sqlnet.ora` file must contain the ENCRYPTION_WALLET_LOCATION parameter formatted exactly as follows:

```
ENCRYPTION_WALLET_LOCATION=(SOURCE=(METHOD=FILE)(METHOD_DATA=
(DIRECTORY=/opt/oracle/dcs/commonstore/wallets/tde/$ORACLE_UNQNAME)))
```



Important

In this wallet location entry, `$ORACLE_UNQNAME` is an environment variable and should not be replaced with an actual value.

To check the TDE wallet location specification

Use the `cat` command to check the TDE wallet location specification. For example:

```
[oracle@orcl tde]$ cat $ORACLE_HOME/network/admin/sqlnet.ora

ENCRYPTION_WALLET_LOCATION=(SOURCE=(METHOD=FILE)(METHOD_DATA=
(DIRECTORY=/opt/oracle/dcs/commonstore/wallets/tde/$ORACLE_UNQNAME)))
```

INCORRECT STATE OF THE TDE WALLET

Database backups fail if the TDE wallet is not in the proper state. The following scenarios can cause this problem:

The ORACLE_UNQNAME environment variable was not set when the database was started using SQL*Plus

If the database was started using SQL*Plus, and the `ORACLE_UNQNAME` environment variable was not set, the wallet is not opened correctly.

To fix the problem, start the database using the `srvctl` utility:

```
srvctl start database -d <db_unique_name>
```

A pluggable database was added with an incorrectly configured master encryption key

In a multitenant environment, each pluggable database (PDB) has its own master encryption key, which is stored in a single keystore used by all containers. After you create or plug in a new PDB, you must create and activate a master encryption key for it. If you do not do so, the `STATUS` column in the `v$encryption_wallet` view shows the value `OPEN_NO_MASTER_KEY`.

To check the master encryption key status and create a master key, do the following:

1. Review the the `STATUS` column in the `v$encryption_wallet` view, as shown in the following example:

```
SQL> alter session set container=pdb2;

Session altered.

SQL> select WRL_TYPE,WRL_PARAMETER,STATUS,WALLET_TYPE from v$encryption_wallet;

WRL_TYPE        WRL_PARAMETER                                          STATUS
WALLET_TYPE

--------------- ------------------------------------------------------ ------------------ ------
---

FILE            /opt/oracle/dcs/commonstore/wallets/tde/example_iadxyz/ OPEN_NO_MASTER_KEY
AUTOLOGIN
```

2. Confirm that the PDB is in READ WRITE open mode and is not restricted, as shown in the following example:

```
SQL> show pdbs

CON_ID CON_NAME     OPEN MODE              RESTRICTED

------ ------------ ---------------------- ---------------

2      PDB$SEED     READ ONLY              NO

3      PDB1         READ WRITE             NO

4      PDB2         READ WRITE             NO
```

   The PDB cannot be open in restricted mode (the `RESTRICTED` column must show `NO`). If the PDB is currently in restricted mode, review the information in the PDB_PLUG_IN_VIOLATIONS view and resolve the issue before continuing. For more information on the PDB_PLUG_IN_VIOLATIONS view and the restricted status, review the documentation on pluggable database for your Oracle database version.

3. Run the following `DBCLI` commands to change the status to `OPEN`:

```
$ sudo su -
 # dbcli list-database
 # dbcli update-tdekey -i <database_ID> -n <PDB_name> -p
```

The `update-tdekey` command shown will prompt you for the admin password.

a. Confirm that the status of the wallet has changed from OPEN_NO_MASTER_KEY to OPEN by querying the `v$encryption_wallet` view as shown in step 1.

INCORRECT CONFIGURATION RELATED TO THE TDE WALLET

Several configuration parameters related to the TDE wallet can cause backups to fail.

To check configuration related to the TDE wallet

- Check that the environment's database unique name parameter (ORACLE_UNQNAME) is set correctly using the following command:

```
srvctl getenv database -d <db_unique_name>
```

For example:

```
[oracle@orcl tde]$ srvctl getenv database -d orclbkp_iadxyz

orclbkp_iadxyz:

ORACLE_UNQNAME=orclbkp_iadxyz

TZ=UTC
```

- Check your `sqlnet.ora` settings to confirm that the file has an ENCRYPTION_WALLET_LOCATION parameter with the correct DIRECTORY value. Use the following command: `cat $ORACLE_HOME/network/admin/sqlnet.ora`

For example:

```
[oracle@orcl tde]$ cat $ORACLE_HOME/network/admin/sqlnet.ora

ENCRYPTION_WALLET_LOCATION=(SOURCE=(METHOD=FILE)(METHOD_DATA=
(DIRECTORY=/opt/oracle/dcs/commonstore/wallets/tde/$ORACLE_UNQNAME)))
```

- Confirm that the wallet status is **open** and the wallet type is **auto login** by checking the `v$encryption_wallet` view. For example:

```
SQL> select status, wrl_parameter,wallet_type from v$encryption_wallet;
```

```
STATUS                         WRL_PARAMETER                                              WALLET_TYPE

------------------------------ ---------------------------------------------------------- ------------
--------

OPEN                           /opt/oracle/dcs/commonstore/wallets/tde/example_iadxyz/ AUTOLOGIN
```

For pluggable databases (PDBs), be sure that you switch to the appropriate container before querying `v$encryption_wallet` view. For example:

```
[oracle@paulo ~]$ sqlplus / as sysdba

SQL> alter session set container=pdb1;

Session altered.

SQL> select WRL_TYPE,WRL_PARAMETER,STATUS,WALLET_TYPE from v$encryption_wallet;

WRL_TYPE   WRL_PARAMETER                                          STATUS     WALLET_TYPE

---------- ------------------------------------------------------ ---------- -------------------
--

FILE       /opt/oracle/dcs/commonstore/wallets/tde/tiger_iad3c8/     OPEN      AUTOLOGIN
```

MISSING TDE WALLET FILE

The TDE wallet file (`ewallet.p12`) can cause backups to fail if it is missing, or if it has incompatible file system permissions or ownership. Check the file as shown in the following example:

```
[oracle@orcl tde]$ ls -ltr /opt/oracle/dcs/commonstore/wallets/tde/$ORACLE_UNQNAME/ewallet.p12

-rwx------ 1 oracle oinstall 5680 Apr 18 13:09 /opt/oracle/dcs/commonstore/wallets/tde/orclbkp_
iadxzy/ewallet.p12
```

The TDE wallet file should have file permissions with the octal value "700" (`-rwx------`), and the owner of this file should be a part of the `oinstall` operating system group.

MISSING AUTO LOGIN WALLET FILE

The auto login wallet file (`cwallet.sso`) can cause backups to fail if it is missing, or if it has incompatible file system permissions or ownership. Check the file as shown in the following example:

```
[oracle@orcl tde]$ ls -ltr /opt/oracle/dcs/commonstore/wallets/tde/$ORACLE_UNQNAME/cwallet.sso

-rwx------ 1 oracle oinstall 5725 Apr 18 13:09 /opt/oracle/dcs/commonstore/wallets/tde/orclbkp_
iadxyz/cwallet.sso
```

The auto login wallet file should have file permissions with the octal value "700" (`-rwx------`), and the owner of this file should be a part of the `oinstall` operating system group.

Other Causes of Backup Failures

UNMOUNTED COMMONSTORE MOUNT POINT

The mount point `/opt/oracle/dcs/commonstore` must be mounted, or backups will fail.

To check the commonstore mount point

Confirm that the mount point `/opt/oracle/dcs/commonstore` is mounted, as shown in the following example:

```
[root@orcl ~]# srvctl config filesystem -volume commonstore -diskgroup data

Volume device: /dev/asm/commonstore-5

Diskgroup name: data

Volume name: commonstore

Canonical volume device: /dev/asm/commonstore-5

Accelerator volume devices:

Mountpoint path: /opt/oracle/dcs/commonstore

Mount point owner: oracle
```

```
Mount users:

Type: ACFS
```

To confirm that ora.data.commonstore.acfs is online

The state for ora.data.commonstore.acfs must be online, or backups will fail. Confirm as shown in the following example:

```
[root@orcl ~]# crsctl stat resource ora.data.commonstore.acfs -v

NAME=ora.data.commonstore.acfs

TYPE=ora.acfs.type

LAST_SERVER=orcl

STATE=OFFLINE

TARGET=OFFLINE
...

STATE_DETAILS=admin unmounted /opt/oracle/dcs/commonstore

...

[root@orcl ~]# ls -ltr /opt/oracle/dcs/commonstore

total 0
```

If the STATE_DETAILS value is `unmounted`, mount the file system as shown in the following example:

```
[root@orcl ~]# srvctl start filesystem -volume commonstore -diskgroup data
```

Confirm that the change was successful as shown in the following example:

```
[root@orcl ~]# crsctl stat resource ora.data.commonstore.acfs -v

NAME=ora.data.commonstore.acfs
```

```
TYPE=ora.acfs.type

LAST_SERVER=orcl

STATE=ONLINE on orcl

TARGET=ONLINE

CARDINALITY_ID=ONLINE

...

STATE_DETAILS=mounted on /opt/oracle/dcs/commonstore
```

List the contents of the `commonstore` directory to confirm that it is mounted, as shown in the following example:

```
[root@orcl ~]# ls -ltr /opt/oracle/dcs/commonstore

total 220

drwx------ 2 root   root     65536 Apr 18 10:50 lost+found

drwx------ 3 oracle oinstall 20480 Apr 18 11:02 wallets

drwxr-xr-x 3 root   root     20480 Apr 20 06:41 pkgrepos

drwxr-xr-x 4 oracle oinstall 20480 Apr 20 06:41 objectstore
```

THE DATABASE IS NOT PROPERLY REGISTERED

Database backups fail if the database is not registered with the dcs-agent. This scenario can occur if you manually migrate the database to Oracle Cloud Infrastructure and do not run the `dbcli register-database` command.

To check whether the database is properly registered, review the information returned by running the `srvctl config database` command and the `dbcli list-databases` command. If either command does not return a record of the database, contact Oracle Support Services.

For instructions on how to register the database, refer to the following topics:

- Registering the Database on the DB System
- dbcli register-database

Obtaining Further Assistance

If you were unable to resolve the problem using the information in this topic, follow the procedures below to collect relevant database and diagnostic information. After you have collected this information, contact Oracle Support.

To collect database information for use in problem reports

Use the following commands to collect details about your database. Record the output of each command for reference:

```
dbcli list-databases
```

```
dbcli describe-database -i <database_id>
```

```
dbcli describe-component
```

To collect diagnostic information regarding failed jobs

1. Log on to the host as the root user and navigate to the `/opt/oracle/dcs/bin/` directory.
2. Run the following two commands to generate information about the failed job:

   ```
   dbcli list-jobs |grep -i <dbname>
   ```

   ```
   dbcli describe-job -i <job_ID> -j
   ```

   The *<job_ID>* in the second command should be the ID of the latest failed job reported from the first command.
3. Run the diagnostics collector script to create a zip file with the diagnostic information for Oracle Support Services.

   ```
   diagcollector.py
   ```

This command creates a file named `diagLogs-<timestamp>.zip` in the `/tmp` directory.

To collect DCS agent log files

To collect DCS agent log files, do the following:

1. Log in as opc user.
2. Run the following command:

```
sudo /opt/oracle/dcs/bin/diagcollector.py
```

3. The system returns a message indicating that agent logs are available in a zip file at a specified directory. For example:

```
[opc@prodpr ~]$ sudo /opt/oracle/dcs/bin/diagcollector.py

Log files collected to :/tmp/dcsdiag/diagLogs-1234567890.zip

Logs are being collected to:

/tmp/dcsdiag/diagLogs-1234567890.zip
```

To collect TDE configuration details

1. Run the `srvctl getenv database -d <db_unique_name>` command and record the output for reference.
2. Record the output of the view `v$encryption_wallet`. For example:

```
SQL> select status, wrl_parameter,wallet_type from v$encryption_wallet;

STATUS          WRL_PARAMETER                                          WALLET_TYPE

--------------- ------------------------------------------------------ ---------

OPEN            /opt/oracle/dcs/commonstore/wallets/tde/example_iadxyz/ AUTOLOGIN
```

3. Record the output of the the output of the `ls -ltr <wrl_parameter>` command. For

example:

```
[oracle@patchtst ~]$ ls -ltr /opt/oracle/dcs/commonstore/wallets/tde/example_iadxyz/

total 28

-rw------- 1 oracle asmadmin 2400 May  2 09:42 ewallet_2018050209420381_defaultTag.p12

-rw------- 1 oracle asmadmin 5680 May  2 09:42 ewallet.p12

-rw------- 1 oracle asmadmin 5723 May  2 09:42 cwallet.sso
```

To collect the RMAN backup report file

Generate RMAN Backup Report File using the following command:

```
dbcli create-rmanbackupreport -i <db_id> -w detailed -rn <report_name>
```

For example:

```
[root@patchtst ~]# dbcli create-rmanbackupreport -i 57fvwxyz-9dc4-45d3-876b-5f850example -w detailed -rn
bkpreport1
```

Locate the report file using the `dbcli describe-rmanbackupreport -in <report_name>` command. The location of the report is given in output. For example:

```
[root@patchtst ~]# dbcli describe-rmanbackupreport -in bkpreport1

Backup Report details

----------------------------------------------------------------

ID: b55vwxyz-c49f-4af3-a956-acccdexample

Report Type: detailed

Location: Node patchtst: /opt/oracle/dcs/log/patchtst/rman/bkup/example_iadxyz/rman_list_backup_
detail/2018-05-02/rman_list_backup_detail_2018-05-02_11-46-51.0359.log

Database ID: 57fvwxyz-9dc4-45d3-876b-5f850example
```

```
CreatedTime: May 2, 2018 11:46:38 AM UTC
```

Patching Failures

Patching operations can fail for various reasons. Typically, an operation fails because a database node is down, there is insufficient space on the file system, or the database host cannot access the object store.

This topic includes information to help you determine the cause of the failure and fix the problem. The information is organized into several sections, based on the error condition. If you already know the cause, you can skip to the section with the suggested solution. Otherwise, use the procedure in Determining the Problem to get started.

Determining the Problem

In the Console, you can identify a failed patching operation by viewing the patch history of a DB system or an individual database. A patch that was not successfully applied displays a status of **Failed** and includes a brief description of the error that caused the failure. If the error message does not contain enough information to point you to a solution, you can use the database CLI and log files to gather more data. Then, refer to the applicable section in this topic for a solution.

To identify the root cause of the patching operation failure

1. Log on to the host as the root user and navigate to the `/opt/oracle/dcs/bin/` directory.
2. Determine the sequence of operations performed on the database.

   ```
   dbcli list-jobs
   ```

   Note the last job ID listed with a status other than **Success**.
3. With the job ID you noted from the previous step, use the following command to check the details of that job:

```
dbcli describe-job -i <job_ID> -j
```

Typically, running this command is enough to reveal the root cause of the failure.

4. If you require more information, review the `/opt/oracle/dcs/log/dcs-agent.log` file.
   You can find the job ID in this file by using the timestamp returned by the job report in step 2.



Note

If the patching failure is on a 2-node RAC database, perform steps 3 and 4 on both nodes.

Database Service Agent Issues

Your Oracle Cloud Infrastructure Database makes use of an agent framework to allow you to manage your database through the cloud platform.

Resolving Patching Failures Caused By a Stopped Agent

Occasionally you might need to restart the dcsagent program if it has the status of **stop/waiting** to resolve a patching failure.

To restart the database service agent

1. From a command prompt, check the status of the agent:

```
initctl status initdcsagent
```

2. If the agent is in the **stop/waiting** state, try to restart the agent:

```
initctl start initdcsagent
```

3. Check the status of the agent again to confirm that it has the **start/running** status:

```
initctl status initdcsagent
```

Resolving Patching Failures Caused By an Agent That Needs to Be Updated

Patching can also fail if your agent needs to be updated. The system gives the following error message for this failure:

```
Current DcsAgent version is less than or equal to minimum required version.
```

To resolve this issue, perform the steps in To have Oracle Support update the Oracle Cloud Infrastructure Database service agent.

To have Oracle Support update the Oracle Cloud Infrastructure Database service agent

1. Confirm that the agent (dcsagent) and DCS Admin program (dcsadmin) are running using the following commands:

   ```
   initctl status initdcsagent
   ```

   ```
   initctl status initdcsadmin
   ```

   If these programs are not running, use the following commands to restart them:

   ```
   initctl start initdcsagent
   ```

   ```
   initctl start initdcsadmin
   ```

2. Follow the instructions in Obtaining Further Assistance to collect your DCS agent log files.
3. Contact Oracle Support for assistance with updating the agent.

Object Store Connectivity Issues

Oracle Cloud Infrastructure DB system and database patches are stored in Oracle Cloud Infrastructure Object Storage. Therefore, successful patching operations require connectivity between the DB system host and the Object Storage location from which the patches are downloaded.

To ensure your database host can connect to Oracle Cloud Infrastructure Object Storage

1. Use the following command to verify the host can access Oracle Cloud Infrastructure Object Storage:

   ```
   dbcli describe-latestpatch
   ```

   Example output indicating success:

   ```
   [root@<host> ~]# dbcli describe-latestpatch

   componentType availableVersion
   --------------- --------------------
   gi 12.2.0.1.180417
   gi 12.1.0.2.180417
   gi 18.2.0.0.180417
   db 11.2.0.4.180417
   db 12.2.0.1.180417
   db 12.1.0.2.180417
   db 18.2.0.0.180417
   oak 12.1.2.11.3
   oak 12.2.1.1.0
   ```

   Example output indicating failure:

   ```
   [root@<host> ~]# dbcli describe-latestpatch

   DCS-10032:Resource patch metadata is not found.Failed to download patchmetadata from
   objectstore
   ```

2. If you cannot connect to the object store, refer to Prerequisites for how to configure object store connectivity.

Host and Oracle Clusterware Issues

One or more of the following conditions on the database host can cause patching operations to fail:

Database Node Not Running During the Patching Operation

All nodes of the database must be active and running while a patching operation is in progress, whether you are patching the DB system or the database home. Use the Console to check that the status of each node is AVAILABLE, and start the node, if needed.

The File System Is Full

Patching operations require a minimum of 15 GB of free space in the `/u01` directory on the host file system. Use the `df -h` command on the host to check the available space. If the file system has insufficient space, you can remove old log or trace files to free up space.

The Oracle Clusterware Is Not Running

Oracle Clusterware enables servers to communicate with each other so that they can function as a collective unit. The cluster software program must be up and running on the DB system for patching operations to complete. Occasionally you might need to restart the Oracle Clusterware to resolve a patching failure.

To restart the Oracle Clusterware

1. From command prompt, check the status of Oracle Clusterware:

   ```
   crsctl check crs
   ```

   Example output:

   ```
   [grid@<host> ~]$ crsctl check crs
   CRS-4638: Oracle High Availability Services is online
   CRS-4537: Cluster Ready Services is online
   CRS-4529: Cluster Synchronization Services is online
   CRS-4533: Event Manager is online
   ```

   For more detailed status information, you can run `crsctl stat res -t`.

2. If Oracle Clusterware is not online, try to restart the program:

   ```
   crsctl start crs
   ```

3. Check the status of Oracle Clusterware to confirm that it is online:

   ```
   crsctl check crs
   ```

The Oracle Grid Infrastructure (GI) Is Not Updated

This problem occurs when you try to patch a database before you patch the DB system of that database. The error description indicates that the Oracle Grid Infrastructure must be updated

first. To resolve this issue, patch the DB system to latest available version. After you patch the DB system, you can retry the database patching operation.

To get the current and latest-available GI versions for the DB system, use the following command:

```
dbcli describe-component
```

Database Issues

An improper database state can lead to patching failures.

Database Not Running During the Patching Operation

The database must be active and running for all of the patching tasks to complete. Otherwise, you must run the datapatch task manually.

To check that the database is active and running

Use the following command to check the state of your database, and ensure that any problems that might have put the database in an improper state are resolved:

```
srvctl status database -d <db_unique_name> -verbose
```

The system returns a message including the database instance status. The instance status must be **Open** for the patching operation to succeed.

If the database is not running, use the following command to start it:

```
srvctl start database -d <db_unique_name> -o open
```

If the database is mounted but does not have the **Open** status, use the following commands to access the SQL*Plus command prompt and set the status to **Open**:

```
sqlplus / as sysdba
```

```
alter database open;
```

To run the datapatch task

Before you run the `datapatch` command, ensure that all pluggable databases (PDBs) are open. To open a PDB, you can use SQL*Plus to execute `ALTER PLUGGABLE DATABASE <pdb_name> OPEN READ WRITE;` against the PDB.

```
$ORACLE_HOME/OPatch/datapatch
```

The datapatch command should be run on each database home.

Obtaining Further Assistance

If you were unable to resolve the problem using the information in this topic, follow the procedures below to collect relevant database and diagnostic information. After you have collected this information, contact Oracle Support.

To collect diagnostic information regarding failed jobs

1. Log on to the host as the root user and navigate to the `/opt/oracle/dcs/bin/` directory.
2. Run the following two commands to generate information about the failed job:

   ```
   dbcli list-jobs
   ```

   ```
   dbcli describe-job -i <job_ID> -j
   ```

   The *<job_ID>* in the second command should be the ID of the latest failed job reported from the first command.
3. Run the diagnostics collector script to create a zip file with the diagnostic information for Oracle Support Services.

   ```
   diagcollector.py
   ```

   This command creates a file named `diagLogs-<timestamp>.zip` in the `/tmp` directory.

To collect DCS agent log files

To collect DCS agent log files, do the following:

1. Log in as opc user.
2. Run the following command:

```
sudo /opt/oracle/dcs/bin/diagcollector.py
```

3. The system returns a message indicating that agent logs are available in a zip file at a specified directory. For example:

```
[opc@prodpr ~]$ sudo /opt/oracle/dcs/bin/diagcollector.py

Log files collected to :/tmp/dcsdiag/diagLogs-1234567890.zip

Logs are being collected to:

/tmp/dcsdiag/diagLogs-1234567890.zip
```

To collect Oracle Grid Infrastructure and Database log files

If an Oracle Grid Infrastructure or Oracle Database patch failed, you can find log files for these failures in the following locations:

Oracle Grid Infrastructure

```
$GI_HOME/cfgtoollogs/
```

Oracle Database

```
$ORACLE_HOME/cfgtoollogs/
```

CHAPTER 10 Edge Services

This chapter explains several services that allow you to manage, secure, and maintain your domains and endpoints.

Overview of the DNS Service

The Oracle Cloud Infrastructure Domain Name System (DNS) service lets you create and manage your DNS zones. You can create zones, add records to zones, and allow Oracle Cloud Infrastructure's edge network to handle your domain's DNS queries.

See Supported Resource Records for additional information.

DNS Service Components

The following list describes the components used to build a DNS zone and make it accessible from the internet.

DOMAIN

Domain names identify a specific location or group of locations on the Internet as a whole. A common definition of "domain" is the complete portion of the DNS tree that has been delegated to a user's control. For example, *example.com* or *oracle.com*.

ZONE

A zone is a portion of the DNS namespace. A Start of Authority record (SOA) defines a zone. A zone contains all labels underneath itself in the tree, unless otherwise specified.

LABEL

Labels are prepended to the zone name, separated by a period, to form the name of a subdomain. For example, the "www" section of www.example.com or the "docs" and "us-ashburn-1" sections of docs.us-ashburn-1.oraclecloud.com are labels. Records are associated with these domains.

CHILD ZONE

Child zones are independent subdomains with their own Start of Authority and Name Server (NS) records. The parent zone of a child zone must contain NS records that refer DNS queries to the name servers responsible for the child zone. Each subsequent child zone creates another link in the delegation chain.

RESOURCE RECORDS

A record contains specific domain information for a zone. Each record type contains information called record data (RDATA). For example, the RDATA of an A or AAAA record contains an IP address for a domain name, while MX records contain information about the mail server for a domain. OCI normalizes all RDATA into the most machine readable format. The returned presentation of your RDATA may differ from its initial input. For more information about RDATA, please see Supported DNS Resource Record Types.

DELEGATION

The name servers where your DNS is hosted and managed.

Ways to Access the DNS Service

You can access Oracle Cloud Infrastructure using the Console (a browser-based interface) or the REST API. Instructions for the Console and API are included in topics throughout this guide.

To access the Console, you must use a supported browser. You can use the Console link at the top of this page to go to the sign-in page. Enter your tenancy, user name, and your password.

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see

Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

DNS Service Capabilities and Limits

The Oracle Cloud Infrastructure DNS service is limited to 1000 zones per account and 25,000 records per zone. Customers with zone and record size needs exceeding these values are encouraged to contact support at support.oracle.com. Zone file uploads are limited to 1 megabyte (MB) in size per zone file. If your zone file is larger than 1 MB, you will need to split the zone file into smaller batches to upload all of the zone information.

Required IAM Service Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

If you're new to policies, see Getting Started with Policies and Common Policies. For more details about policies for DNS, see Details for the DNS Service.

Getting Started with DNS

If you're new to Oracle Cloud Infrastructure DNS, this topic gives guidance on how to proceed.

What is DNS?

The Domain Name System (DNS) translates human-readable domain names to machine-readable IP addresses. A DNS nameserver stores the DNS records for a zone, and responds

with answers to queries against its database. When you type a domain name into your browser, your operating system queries several DNS nameservers until it finds the authoritative nameserver for that domain. The authoritative nameserver then responds with an IP address or other requested record data. The answer is then relayed back to your browser and the DNS record is resolved to the web page.

Creating a Zone

In this step, you will create a zone. A zone holds the trusted DNS records that will reside on Oracle Cloud Infrastructure's nameservers.

To add a zone

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **DNS Zone Management**.
2. Click **Create Zone**.
3. In the **Create Zone** dialog box, choose one of the following methods:
    - **Manual** - Enter the following:
        - **Zone Name**: Enter the name of a zone you want to create. Avoid entering confidential information.
        - **Zone Type**: If you want to control the zone contents directly within Oracle Cloud Infrastructure, select **Primary**. If you want Oracle Cloud Infrastructure to pull zone contents from an external server, select **Secondary** and enter your **Zone Master Server IP** address.
    - **Import** - Drag and drop, select, or paste a valid zone file into the Import Zone File window. The zone is imported as a primary zone. For information about formatting a zone file, see Formatting a Zone File.
4. Click **Submit**.

The system creates and publishes the zone, complete with the necessary SOA and NS records. For more information on adding a record to your zone, see To add a zone record.

Delegating Your Zone

In this step, you will delegate your domain with your registrar. Delegating your domain with your domain's registrar makes your Oracle Cloud Infrastructure hosted zone accessible through the internet.

To delegate a zone

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **DNS Zone Management**.
2. Click the Zone Name for the zone you want to delegate. Zone details and a list of records appear.
3. Use the **Type** sort filter to locate the NS records for your zone.
4. Note the name servers in the RDATA field within each NS record.
5. You can use the noted name servers to change your domain's DNS delegation. Refer to your registrar's documentation for instructions.



Note

Once delegation has completed, allow 24 hours for your delegation to propagate across the internet.

To add a zone record



Tip

There are many record types you can add to your zone, depending on your goals for the zone and its DNS management.

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **DNS Zone Management**.
2. Click the **Zone Name** in which you want to add a record. Zone details and a list of records appear.



Tip

You can use the Zone Name sort filter to list to sort zone names alphanumerically in ascending or descending order.

3. Click **Add Record**.
4. In the **Add Record** dialog box, select a record type from the drop-down list, and then enter the information for the record. Avoid entering confidential information. For more information about record types, see Supported Resource Records.
5. (Optional) Click the **Add Another Record** check box to add multiple records in succession.
6. Click **Submit**.
7. Once your records have been added, click **Publish Changes**.
8. In the confirmation dialog box, click **Publish Changes**.

Common DNS Zone Record Types

For a complete list of records supported by Oracle Cloud Infrastructure DNS, see Supported Resource Records.

A

An address record used to point a hostname to an IPv4 address. For more information about A records, see RFC 1035.

AAAA

An address record used point a hostname at an IPv6 address. For more information about AAAA records, see RFC 3596.

CNAME

A Canonical Name record identifies the canonical name for a domain. For more information about CNAME records, see RFC 1035.



Note

Per RFC 1912, CNAMES cannot be placed at the apex of the zone.

MX

A Mail Exchanger record defines the mail server accepting mail for a domain. MX records must point to a hostname. MX records must not point to a CNAME or IP address. For more information about MX records, see RFC 1035.

TXT

A Text record holds descriptive, human readable text, and can also include non-human readable content for specific uses. It is commonly used for SPF records and DKIM records that require non-human readable text items. For more information about TXT records, see RFC 1035.

Testing DNS Using BIND's dig Tool

Using the Domain Information Groper (dig) command line tool, you can test against the delegation where your domain is hosted, and you will immediately see whether the change took place without accounting for the cache or TTL (Time to Live) that you have configured.

For more information on using dig to test your DNS, see Testing DNS Using BIND'S dig Tool.

Managing DNS Service Zones

The Oracle Cloud Infrastructure DNS service enables you to manage zones and view zone reports within the Console.

Using the Console

MANAGING ZONES AND ZONE RECORDS

To add a zone

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **DNS Zone Management**.
2. Click **Create Zone**.
3. In the **Create Zone** dialog box, choose one of the following methods:
    - **Manual** - Enter the following:
        - **Zone Name**: Enter the name of a zone you want to create. Avoid entering confidential information.
        - **Zone Type**: If you want to control the zone contents directly within OCI, select **Primary**. If you want OCI to pull zone contents from an external server, select **Secondary** and enter your **Zone Master Server IP** address.
    - **Import** - Drag and drop, select, or paste a valid zone file into the Import Zone File window. The zone is imported as a primary zone. For information about formatting a zone file or how to amend a zone file exported from GoDaddy.com, please see Formatting a Zone File.
4. Click **Submit**.

The system creates and publishes the zone, complete with the necessary SOA and NS records. For more information on adding a record to your zone, see To add a zone record.

To update a secondary zone

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **DNS Zone Management**.
2. Click the secondary **Zone Name** you want to update. Zone details and a list of master server IPs appear.



Tip

You can use the Zone Type sort filter to sort zone type alphanumerically in ascending or descending order.

3. Select the checkbox for the Master Server IP you want to update, and then select **Edit** from the **Actions** drop-down menu.
4. Make the needed changes, and then click **Submit**.
5. (Optional) Click **Add Master Server** to add another Master Server IP address.
6. Click **Publish Changes**.
7. In the confirmation dialog box, click **Publish Changes**.



Tip

For OCI to transfer data from your zone, your nameservers must be able to accept a transfer request from the following IP addresses: 208.78.68.65, 204.13.249.65, 2600:2001:0:1::65, 2600:2003:0:1::65

To delete a zone



Warning

Deletion permanently removes a zone from your DNS service.

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **DNS Zone Management**.
2. Select the checkbox for the zone you want to delete.
3. Click **Delete**. The zone is staged for deletion.
4. Click **Publish Changes** to delete the zone.
5. In the confirmation dialog box, click **Publish Changes**.

To add a zone record



Tip

There are many record types you can add to your zone, depending on your goals for the zone and its DNS management.

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **DNS Zone Management**.
2. Click the **Zone Name** in which you want to add a record. Zone details and a list of records appear.



Tip

You can use the Zone Name sort filter to list to sort zone names alphanumerically in ascending or descending order.

3. Click **Add Record**.
4. In the **Add Record** dialog box, select a record type from the drop-down list, and then enter the information for the record. Avoid entering confidential information. For more information about record types, see Supported Resource Records.
5. (Optional) Click the **Add Another Record** check box to add multiple records in succession.
6. Click **Submit**.
7. Once your records have been added, click **Publish Changes**.
8. In the confirmation dialog box, click **Publish Changes**.

To update a zone record



Note

Protected Records

You can change various components of the records within your zones, such as time-to-live (TTL) and



relevant RDATA. However, some records contain information that cannot be changed. A lock symbol indicates a protected record. You can attempt changes to such records through the **Actions** menu, but the system might not permit updates to some fields.

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **DNS Zone Management**.
2. Click the **Zone Name** in which you want to update a record. Zone details and a list of records appear.



Tip

You can use the Zone Name sort filter to sort zone names alphanumerically in ascending or descending order.

3. To help find a record, you can use the following filter options:
   - Enter the name of the record's domain in the **Search** field.
   - To find unpublished records, select the **Staged** check box.
   - To find published records, select the **Unstaged** check box.
   - Use the **Domain**, **TTL**, or **Type** sort filter to sort records.
4. Select the checkbox for the record you want to update, and select **Edit** from the **Actions** drop-down menu.
5. In the **Edit Record** dialog box, make the needed changes, and then click **Submit**.
6. Click **Publish Changes**.
7. In the confirmation dialog box, click **Publish Changes**.

Reverting Changes Before Publishing

You can revert records to their current published state before you publish changes. Once a record has been published, it cannot be reverted. Select the checkbox for the record you want to revert, and then select **Revert** from the **Actions** drop-down menu.

To delete a zone record

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **DNS Zone Management**.
2. Click the **Zone Name** in which you want to delete a record. Zone details and a list of records appear.



Tip

You can use the Zone Name sort filter to sort zone names alphanumerically in ascending or descending order.

3. Select the checkbox for the record you want to delete, and then select **Delete** from the **Actions** drop-down menu.
4. Click **Publish Changes**.
5. In the confirmation dialog box, click **Publish Changes**.

To delegate a zone

To make your Oracle Cloud Infrastructure hosted zone accessible through the internet, you must delegate your domain with your domain's registrar.

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **DNS Zone Management**.
2. Click the Zone Name for the zone you want to delegate. Zone details and a list of records appear.
3. Use the **Type** sort filter to locate the NS records for your zone.
4. Note the name servers in the RDATA field within each NS record.
5. You can use the noted name servers to change your domain's DNS delegation. Refer to your registrar's documentation for instructions.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operations to manage your DNS zones:

- GetZone
- ListZones
- CreateZone
- UpdateZone
- DeleteZone
- PatchZoneRecords (add or delete records)
- UpdateZoneRecords

Supported Resource Records

The Oracle Cloud Infrastructure DNS service supports many resource record types. The following list provides a brief explanation of the purpose of each supported record type. Avoid entering confidential information when entering record data. The RFC links direct you to further information about the record types and data structure.

Note About RDATA

OCI normalizes all RDATA into the most machine readable format. The returned presentation of your RDATA may differ from its initial input.

Example:

The RDATA for the ALIAS, CNAME, DNAME, MX, and NS record types may contain one or more absolute domain names. If the specified RDATA for one of these record types does not end in a dot or period to represent the root, the period will be added.

```
www.example.com --> www.example.com.
```

You can use various DNS libraries to normalize your RDATA before input.

Programming Language	Library
Go	DNS Library in Go
Java	dnsjava
Python	dnspython

DNS Resource Record Types

A

An address record used to point a hostname to an IPv4 address. For more information about A records, see RFC 1035.

AAAA

An address record used point a hostname at an IPv6 address. For more information about AAAA records, see RFC 3596.

ALIAS

A private pseudo-record that allows CNAME functionality at the apex of a zone. You can view and read ALIAS records in Oracle Cloud Infrastructure DNS, but you cannot create them.

CAA

A Certification Authority Authorization record allows a domain name holder to specify one or more Certification Authorities authorized to issue certificates for that domain. For more information about CAA records, see RFC 6844.

CDNSKEY

A Child DNSKEY moves a CDNSSEC key from a child zone to a parent zone. The information provided in this record must match the CDNSKEY information for your domain at your other DNS provider. This record is automatically created if you enable DNSSEC on a primary zone in Oracle Cloud Infrastructure DNS. For more information about CDNSKEY, see RFC 7344.

CDS

A Child Delegation Signer record is a child copy of a DS record, for transfer to a parent zone. For more information about CDS records, see RFC 7344.

CERT

A Certificate record stores public key certificates and related certificate revocation lists in the DNS. For more information about CERT records, see RFC 2538 and RFC 4398.

CNAME

A Canonical Name record identifies the canonical name for a domain. For more information about CNAME records, see RFC 1035.

CSYNC

A Child-to-Parent Synchronization record syncs records from a child zone to a parent zone. For more information about CNAME records, see RFC 7477.

DHCID

A DHCP identifier record provides a way to store DHCP client identifiers in the DNS to eliminate potential hostname conflicts within a zone. For more information about DHCID, see RFC 4701.

DKIM

A Domain Keys Identified Mail is a special TXT record set up specifically to supply a public key used to authenticate arriving mail for a domain. For more information about DKIM records, see RFC 6376.

DNAME

A Delegation Name record has similar behavior to a CNAME record, but allows you to map an entire subtree beneath a label to another domain. For more information about DNAME records, see RFC 6672.

DNSKEY

A DNS Key record documents public keys used for DNSSEC. The information in this record must match the DNSKEY information for your domain at your other DNS provider. For more information about DNSKEY records, see RFC 4034.

DS

A Delegation Signer record resides at the top-level domain and points to a child zone's DNSKEY record. DS records are created when DNSSEC security authentication is added to the zone. For more information about DS records, see RFC 4034.

IPSECKEY

An IPSec Key record stores public keys for a host, network, or application to connect to IP security (IPSec) systems. For more information on IPSECKEY records, see RFC 4025.

KEY

A Key record stores a public key that is associated with a domain name. Currently only used by SIG and TKEY records. IPSECKEY and DNSKEY have replaced key for use in IPSec and DNSSEC, respectively. For more information about KEY records, see RFC 4025.

KX

A Key Exchanger record identifies a key management agent for the associated domain name with some cryptographic systems (not including DNSSEC). For more information about KX records, see RFC 2230.

LOC

A Location record stores geographic location data of computers, subnets, and networks within the DNS. For more information about LOC records, see RFC 1876.

MX

A Mail Exchanger record defines the mail server accepting mail for a domain. MX records must point to a hostname. MX records must not point to a CNAME or IP address. For more information about MX records, see RFC 1035.

NS

A Nameserver record lists the authoritative nameservers for a zone. Oracle Cloud Infrastructure DNS automatically generates NS records at the apex of each new primary zone. For more information about NS records, see RFC 1035.

PTR

A Pointer record reverse maps an IP address to a hostname. This behavior is the opposite of an A Record, which forward maps a hostname to an IP address. PTR records are commonly found in reverse DNS zones. For more information about PTR records, see RFC 1035.

PX

A resource record used in X.400 mapping protocols. For more information about PX records, see RFC 822 and RFC 2163.

SOA

A Start of Authority record specifies authoritative information about a DNS zone, including:

- The primary nameserver.
- The email of the domain administrator.

- The domain serial number.
- Several timers relating to refreshing the zone.

The Oracle Cloud Infrastructure DNS automatically generates an SOA record when a zone is created. For more information about SOA records, see RFC 1035.

SPF

A Sender Policy Framework record is a special TXT record used to store data designed to detect email spoofing. For more information about SPF records, see RFC 4408.

SRV

A Service Locator record allows administrators to use several servers for a single domain. For more information about SRV records, see RFC 2782.

SSHFP

An SSH Public Key Fingerprint record publishes SSH public host key fingerprints using the DNS. For more information about SSHFP records, see RFC 6594.

TLSA

A Transport Layer Security Authentication record associates a TLS server certificate, or public key, with the domain name where the record is found. This relationship is called a TLSA certificate association. For more information about TLSA records, see RFC 6698.

TXT

A Text record holds descriptive, human readable text, and can also include non-human readable content for specific uses. It is commonly used for SPF records and DKIM records that require non-human readable text items. For more information about TXT records, see RFC 1035.

How to Format a Zone File

A zone file is a text file that describes a DNS zone. The BIND file format is the industry preferred zone file format and has been widely adopted by DNS server software. The format

is defined in RFC 1035.

Example of an OCI Zone File

This an example of a zone file downloaded from OCI's DNS service.

```
$ORIGIN example.com.
                        @                    3600 SOA   ns1.p30.oraclecloud.net. (
                        zone-admin.dyndns.com.     ; address of responsible party
                        2016072701                 ; serial number
                        3600                       ; refresh period
                        600                        ; retry period
                        604800                     ; expire time
                        1800                     ) ; minimum ttl
                        86400 NS    ns1.p68.dns.oraclecloud.net.
                        86400 NS    ns2.p68.dns.oraclecloud.net.
                        86400 NS    ns3.p68.dns.oraclecloud.net.
                        86400 NS    ns4.p68.dns.oraclecloud.net.
                        3600 MX    10 mail.example.com.
                        3600 MX    20 vpn.example.com.
                        3600 MX    30 mail.example.com.
                        60 A     204.13.248.106
                        3600 TXT   "v=spf1 includespf.oraclecloud.net ~all"
                        mail                  14400 A     204.13.248.106
                        vpn                      60 A     216.146.45.240
                        webapp                   60 A     216.146.46.10
                        webapp                   60 A     216.146.46.11
                www                   43200 CNAME example.com.
```



Note

Record Classes

In the example zone file above, no record classes (IN, CH, HS) are displayed. OCI’s DNS service only works with Internet (IN) class records but omits the class information in zone files for efficiency purposes.

Anatomy of a Zone File

`$ORIGIN` indicates a DNS node tree and will typically start a DNS zone file. Any host labels below the origin will append the origin hostname to assemble a fully qualified hostname. Any host label within a record that uses a fully qualified domain terminating with an ending period will not append the origin hostname.

Example: With `$ORIGIN example.com.`, any record where the host label field is not followed by a period, `example.com`. will be appended to them.

The "@" symbol is a special label that indicates the `$ORIGIN` should replace the "@" symbol. This is typically used for the apex of a zone.

SOA Record – The `$ORIGIN` is followed by the zone's Start Of Authority (SOA) record. An SOA record is required for each zone. It contains the name of the zone, the e-mail address of the party responsible for administering the domain's zone file, the current serial number of the zone, the primary nameserver of the zone, and various timing elements (measured in seconds).

SOA Record Format

```
@     IN     SOA    {primary-name-server}     {hostmaster-email} (
                           {serial-number}
                           {time-to-refresh}
                           {time-to-retry}
                           {time-to-expire}
                      {minimum-TTL} )
```

- **Primary Name Server** – The nameserver that contains the original zone file and not an AXFR transferred copy.
- **Hostmaster Email** – Address of the party responsible for the zone. A period "." is used in place of an "@" symbol. For email addresses that contain a period, this will be escaped with a slash "/".
- **Serial Number** – Version number of the zone. The serial number will increase with each subsequent update to your zone.
- **Time To Refresh** – How long a nameserver should wait prior to checking for a serial number increase within the primary zone file, in seconds. An increased serial number

detected by a secondary DNS nameserver means a transfer is needed to sync your records. Only applies to zones using secondary DNS.

- **Time To Retry** – How long a nameserver should wait prior to retrying to update a zone after a failed attempt, in seconds. Only applies to zones using secondary DNS.
- **Time To Expire** – How long a nameserver should wait prior to considering data from a secondary zone invalid and stop answering queries for that zone, in seconds. Only applies to zones using secondary DNS.
- **Minimum TTL** – Minimum Time To Live (TTL). How long a nameserver or resolver should cache a negative response, in seconds.

Anatomy of a Record Within a Zone File

A zone file is a collection of resource records with each record entry described in the following sequence:

Format:	Host Label	TTL	Record Class	Record Type	Record Data
Example:	example.com.	60	IN	A	104.255.228.125

- **Host Label** – A host label helps to define the hostname of a record and whether the `$ORIGIN` hostname will be appended to the label. Fully qualified hostnames terminated by a period will not append the origin.
- **TTL** – The Time To LIve (TTL) is the amount of time that a DNS record will be cached by an outside DNS server or resolver, in seconds.
- **Record Class** – There are three classes of DNS records: IN (Internet), CH (Chaosnet), and HS (Hesiod). OCI DNS only uses the IN class of records.
- **Record Type** – Where the format of a record is defined.
- **Record Data** – The data within a DNS answer, such as an IP address, hostname, or other information. Different record types will contain different types of record data.

Amending Zone Files Exported from GoDaddy.com for Import

GoDaddy.com exports zone files in a proprietary format. To get the Oracle Cloud Infrastructure DNS service to correctly import a zone file exported from GoDaddy.com, you must directly alter the file. Follow these instructions to update the zone file.

1. Export your zone file from GoDaddy.com. Reference GoDaddy.com's documentation to see how this is done.
2. Open the file in your preferred text editor.
3. Prepend a new line to the file before the SOA record with the following information, including the trailing period: `$ORIGIN [yourdomain].`
4. Once the file has been amended, save the changes to the file and use the zone import function to import the file into your DNS configuration. For more information about zone import, see Managing DNS Zones.



Note

If your zone file includes includes dynamic A records, such as `@ 600 IN A GoCentral Published Site`, you will need to amend these records with the correct IP addresses of your website. Please contact GoDaddy.com for information about how to obtain this information.

Example: `@ 600 IN A 192.0.2.255`

Example:

This is an example of a zone file exported from GoDaddy.com. The code in bold is the code that needs to be removed from the file for it to be eligible for import into Oracle Cloud Infrastructure DNS.



Tip

Placing a semi-colon at the beginning of a line is valid comment syntax for a zone file, per RFC 1035, but for ease of use and formatting it is recommended to remove the large section of comments from the beginning of the zone file provided by GoDaddy.com, as shown below.

```
Domain: example.com
; Exported (y-m-d hh:mm:ss): 2019-01-10 13:05:04
;
; This file is intended for use for informational and archival
; purposes ONLY and MUST be edited before use on a production
; DNS server.
;
; In particular, you must update the SOA record with the correct
; authoritative name server and contact e-mail address information,
; and add the correct NS records for the name servers which will
; be authoritative for this domain.
;
; For further information, please consult the BIND documentation
; located on the following website:
;
; http://www.isc.org/
;
; And RFC 1035:
;
; http://www.ietf.org/rfc/rfc1035.txt
;
; Please note that we do NOT offer technical support for any use
; of this zone data, the BIND name server, or any other third-
; party DNS software.
;
; Use at your own risk.
; SOA Record
example.com.        3600     IN     SOA ns41.domaincontrol.com. dns.net. (
                    2018122702
```

```
                    28800
                    7200
                    604800
                    3600
                    )
; A Records
@   600  IN     A   192.0.2.249
blog    10800    IN     A   192.0.2.255
dev 1800     IN     A   192.0.2.254
dev01   1800     IN     A   192.0.2.253
dev02   1800     IN     A   192.0.2.252
dev03   1800     IN     A   192.0.2.251
dev04   1800     IN     A   192.0.2.250
; CNAME Records
abc123b432dc7785b7ef31f04f25c3e71    1800     IN     CNAME   verify.bing.com.
akamai  600  IN     CNAME   www.example.com.edgekey.net.
email   3600     IN     CNAME   email.secureserver.net.
; MX Records
@   604800   IN     MX  10  amlxe.l.google.com.
@   604800   IN     MX  10  aplxe.l.google.com.
; TXT Records
@   3600     IN     TXT "google-site-verification=3J82-80dbMyCo5Q5C1G11JszeOnZPGCSYlHcPcXg"
@   3600     IN     TXT "google-site-verification=eS_QPYLE_W4nduSrlN-cddxG7ZqOnB743xsbX918"
```

Below is an example of an amended zone file ready to import into Oracle Cloud Infrastructure DNS. The code in bold needs to be prepended to your zone file before import.

```
$ORIGIN example.com.
example.com.        3600     IN     SOA ns41.domaincontrol.com. dns.net. (
                    2018122702
                    28800
                    7200
                    604800
                    3600
                    )

; A Records
@   600  IN     A   192.0.2.249
blog    10800    IN     A   192.0.2.255
dev 1800     IN     A   192.0.2.254
```

```
dev01   1800     IN     A   192.0.2.253
dev02   1800     IN     A   192.0.2.252
dev03   1800     IN     A   192.0.2.251
dev04   1800     IN     A   192.0.2.250abc123b432dc7785b7ef31f04f25c3e71    1800     IN     CNAME
verify.bing.com.
; CNAME Records
akamai  600  IN     CNAME   www.example.edgekey.net.
email   3600     IN     CNAME   email.secureserver.net.
; MX Records
@   604800   IN     MX  10  amlxe.l.google.com.
@   604800   IN     MX  10  aplxe.l.google.com.
; TXT Records
@   3600     IN     TXT "google-site-verification=3J82-80dbMyCo5Q5C1GM8os1VYVEOnZPGCSYlHcPcXg"
@   3600     IN     TXT "google-site-verification=eS_QPYLE_W4nduSrlN-cddxG7ZqOnB7k7uIG7qrsyu8"
```

Testing DNS Using BIND'S dig Tool

Using the Domain Information Groper (dig) command line tool, you can test against the delegation where your domain is hosted, and you will immediately see whether the change took place without accounting for the cache or TTL (Time to Live) that you have configured.



Note

Windows users can download the tool from BIND's website. Use Terminal to access dig on Linux and Macintosh systems.

Using dig

Before using BIND's dig tool, you must access or install dig on your system. Once you have access to dig, you can use dig to test your DNS.

To access dig (Mac)

1. From your Applications folder, open the Utilities folder, and then select **Terminal**.

2. When Terminal is open, type a dig command using a hostname you want to look up.

To Install dig (Windows)

1. Go to BIND's website and download the most current, stable version of BIND.



Note

BIND supports both 32 and 64 bit Windows systems. Confirm which version of Windows you are using and download the correct version of BIND. View Microsoft's documentation to determine which version of Windows you are using.

2. Extract the downloaded file and install BIND in the following directory: **C:\Program Files\ISC BIND 9**. Select the **Tools Only** check box.
3. Once BIND is installed, on the Windows menu open the Control Panel, and then open your System properties.
4. On the **Advanced** tab, click **Environment Variables**.
5. Under **System Variables**, select **Path**, and then click **Edit**.
6. At the end of the path in the Edit System Variable window, add **C:\Program Files\ISC BIND 9\bin**, and then click **OK**.
7. In the Edit Variables window, click **OK**. In the System properties window, click **OK**.

To open the Command Prompt

For Windows versions 8 -10:

1. Click the Windows menu icon.
2. In the **Search** field, type **CMD**.
3. Click **Command Prompt**.

For Windows version 7:

1. On the **Start** menu click **Run**.
2. Enter **CMD**, and then click **OK**.

To use dig to test your DNS

1. Open Terminal (Mac and Linux) or Command Prompt (Windows).
2. Type `dig <any hostname>`, and then press **Enter**.

The following information is returned:

```
$ dig oracle.com

; <<>> DiG 9.10.6 <<>> oracle.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 45619
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 6, ADDITIONAL: 13

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;oracle.com.                    IN      A

;; ANSWER SECTION:
oracle.com.             300     IN      A       137.254.16.101

;; AUTHORITY SECTION:
oracle.com.             10800   IN      NS      dnsmaster5.oracle.com.
oracle.com.             10800   IN      NS      dnsmaster6.oracle.com.
oracle.com.             10800   IN      NS      dnsmaster3.oracle.com.
oracle.com.             10800   IN      NS      dnsmaster1.oracle.com.
oracle.com.             10800   IN      NS      dnsmaster4.oracle.com.
oracle.com.             10800   IN      NS      dnsmaster2.oracle.com.

;; ADDITIONAL SECTION:
dnsmaster5.oracle.com.  10800   IN      A       192.135.82.70
dnsmaster4.oracle.com.  10800   IN      A       192.135.82.52
dnsmaster3.oracle.com.  10800   IN      A       192.135.82.36
dnsmaster6.oracle.com.  10800   IN      A       192.135.82.84
dnsmaster2.oracle.com.  10800   IN      A       10.221.8.13
dnsmaster1.oracle.com.  10800   IN      A       192.135.82.4
dnsmaster5.oracle.com.  10800   IN      AAAA    2606:b400:1400:4240:4fff:ffff:ffff:9f99
dnsmaster4.oracle.com.  10800   IN      AAAA    2606:b400:1400:8140:4fff:ffff:ffff:9f99
dnsmaster3.oracle.com.  10800   IN      AAAA    2606:b400:1400:8040:4fff:ffff:ffff:9f99
dnsmaster6.oracle.com.  10800   IN      AAAA    2606:b400:1400:4144::41
dnsmaster2.oracle.com.  10800   IN      AAAA    2606:b400:1400:280:feed::3
dnsmaster1.oracle.com.  10800   IN      AAAA    2606:b400:1400:180:4fff:ffff:ffff:9f99

;; Query time: 163 msec
;; SERVER: 192.135.82.52#53(192.135.82.52)
;; WHEN: Tue Feb 26 14:02:05 EST 2019
;; MSG SIZE  rcvd: 469
```

- **Question section**: The query made to the DNS. In this example, we asked for the first available A record for the hostname, oracle.com.

- **Answer section**: The first available answer for the query made to the DNS. In this example, we received the A record for the IP address 137.254.16.101.
- **Authority section**: The authoritative nameservers from which the answer to the query was received. These nameservers house the zones for a domain.
- **Additional section**: Additional information the resolver may need but not the answer to the query.

DIG COMMANDS

Command	Description	Example
`dig [hostname]`	Returns any A record found within the queried hostname's zone.	`dig oracle.com`
`dig [hostname] [record type]`	Returns the records of that type found within the queried hostname's zone.	`dig oracle.com MX`
`dig [hostname] +short`	Provides a brief answer, usually just an IP address.	`dig oracle.com +short`
`dig @[nameserver address] [hostname]`	Queries the nameserver directly instead of your ISP's resolver.	`dig @dnsmaster6.oracle.com`
`dig [hostname] +trace`	Adding `+trace` instructs dig to resolve the query from the root nameserver downwards and to report the results from each query step.	`dig dyn.com +trace`

Command	Description	Example
`dig -X [IP address]`	Reverse lookup for IP addresses.	`dig -X 137.254.16.101`
`dig [hostname] any`	Returns all records for a hostname.	`dig oracle.com any`

Overview of the Health Checks Service

The Oracle Cloud Infrastructure Health Checks service provides users with high frequency external monitoring to determine the availability and performance of any publicly facing service, including hosted websites, API endpoints, or externally facing load balancers. By using Health Checks, users can ensure that they are immediately aware of any availability issue affecting their customers.

Health Checks Service Components

The following list describes the key components used in creating a health check.

MONITORS

Monitors allow you to continuously monitor the health of public-facing endpoints. You can configure monitors to use either HTTP and ping protocols.

ON-DEMAND PROBES

On-demand probes allow you to execute a one-time probe to assess the health of a public-facing endpoint. You can configure on-demand probes to use either or both HTTP and ping protocols. This feature is currently only available via the REST API.

VANTAGE POINTS

Vantage points are geographic locations from which monitors and probes can be executed to your specified target. Oracle Cloud Infrastructure maintains dozens of vantage points around the world.

PROTOCOLS

The Health Checks service allows you to configure both HTTP and ping type monitors. Each type has respective protocols.

Ways to Access the Health Checks Service

You can access Oracle Cloud Infrastructure using the Console (a browser-based interface) or the REST API. Instructions for the Console and API are included in topics throughout this guide.

To access the Console, you must use a supported browser. You can use the Console link at the top of this page to go to the sign-in page. Enter your tenancy, user name, and your password.

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

Health Checks Service Capabilities and Limits

The Oracle Cloud Infrastructure Health Checks service is limited to 1000 endpoint tests per account.

See Service Limits for a list of applicable limits and instructions for requesting a limit increase.

Required IAM Service Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

If you're new to policies, see Getting Started with Policies and Common Policies. For more details about policies for Health Checks, see Details for the Health Checks Service.

Policy examples:

- To enable all operations on Health Checks for all users in a tenant :

```
Allow any-user to manage health-check-family in tenancy
```

- To enable all operations on Health Checks for all users in a compartment:

```
Allow any-user to manage health-check-family in compartment <Compartment Name>
```

- To enable all operations on Health Checks for a specific user group:

```
Allow group <Your Group Name> to manage health-check-family in compartment <Compartment Name>
```

Tagging Resources

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

Getting Started With the Health Checks API

The Health Checks service allows you to configure and deploy monitors and on-demand probes using the Health Checks API. Use the following guide to learn how to set up monitors and probes, and retrieve their results using the REST API.

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

Endpoints

The Health Checks API can be accessed via the following endpoints:

- https://healthchecks.ca-toronto-1.oraclecloud.com/20180501
- https://healthchecks.us-ashburn-1.oraclecloud.com/20180501
- https://healthchecks.us-phoenix-1.oraclecloud.com/20180501
- https://healthchecks.eu-frankfurt-1.oraclecloud.com/20180501
- https://healthchecks.uk-london-1.oraclecloud.com/20180501

Available Protocols For Probes and Monitors

You can configure monitors and probes to use HTTP or ping requests. You will need to ensure that the endpoint being monitored is configured to accept the specified protocol.

HTTP - Configure a GET or HEAD request using HTTP/1.1 to test the target for availability. The probe results are returned in JSON and include the HTTP Status Code and DNS lookup, connection and response timings.

HTTPS - Configure an encrypted HTTPS GET or HEAD request to test the availability of any secure hosted target. Defaults to port 443. The probe results are returned in JSON and include the HTTP Status Code and DNS lookup, connection and response timings.

ICMP - Configure an ICMP echo request ping. The results include the round trip time (RTT) latency.

TCP - Configure a TCP handshake to the specified end point. You should be sure to own this endpoint as testing this connection can be costly to the recipient. The results include the round trip time (RTT) latency.

Create A Monitor

Monitors allow you to monitor the health of endpoints over time. The following example shows how to create an HTTPS monitor that checks the health of www.example.com at an interval of every 30 seconds using a GET request.

```
POST /20180501/httpMonitors

                              {
                              "compartmentId":
"ocid1.compartment.oc1..aaaaaaaat7uqcb6zoxvzoga4d4vh4dtweciavepacd3skz56atf3qp73d7fx",
                              "protocol": "HTTPS",
                              "port": 443,
                              "targets": [
                              "www.example.com"
                              ],
                              "timeoutInSeconds": 30,
                              "method": "GET",
                              "displayName": "Example HTTP monitor",
                              "intervalInSeconds": 30
              }
```

Targets can be either hostnames or IP addresses, and the `path` field can be used to specify an optional path, such as www.example.com/project/help.htm. Optionally, you can specify which geographic locations you would like the monitor to launch from by using the `vantagePointNames` field. At least one vantage point must be listed when using this field. For a list of available vantage points, see Vantage Points.

A 200 response will be returned with the successful creation of a probe, and the results of the probe can be retrieved from the URL in the `resultsUrl` field of the response.

```
{
                              "id": "ocid1.httpmonitor.OC2...abuxgljr6l4nepxjkmbtnibwqpu5z24xdvmr7okzoi47wicoflrxh32rwd
                              "compartmentId":
"ocid1.compartment.oc1..aaaaaaacsxgoz7flzmry5dr6scf7xru6lwqpoygqld74npvn5rdj2cv3iiq",
                              "resultsUrl": "https://healthchecks.us-ashburn-
1.oraclecloud.com/20180501/httpProbeResults/ocid1.httpmonitor.OC2...abuxgljr6l4nepxjkmbtnibwqpu5z24xdvm
r7okzoi47wicoflrxh32rwd7a",
                              "targets": [
                              "www.example.com",
                              "www.oracle.com"
                              ],
                              "vantagePointNames": [
                              "ibm-sjc",
                              "aws-dub",
                              "dgo-nyc"
                              ],
                              "protocol": "HTTPS",
                              "timeoutInSeconds": 30,
                              "displayName": "Example Monitor",
                              "intervalInSeconds": 30,
                              "isEnabled": true
               }
```

For more information about creating an HTTP monitor, see CreateHttpMonitor.



Note

You can configure a similar style monitors using TCP or ICMP protocols. For more information, see _ **CreatePingMonitor**.

Create An On-Demand Probe

Probes are one-off health assessments of an endpoint that can be deployed at anytime. The following example shows how to create an on-demand HTTP probe that checks the health of www.example.com with a GET request.

```
POST /20180501/httpProbeResults

{
  "compartmentId":
"ocid1.compartment.oc1..aaaaaaaat7uqcb6zoxvzoga4d4vh4dtweciavepacd3skz56atf3qp73d7fx",
  "protocol": "HTTP",
  "targets": [
      "www.example.com"
  ],
  "timeoutInSeconds": 30,
  "method": "GET"
}
```

Targets can be either hostnames or IP addresses, and the `path` field can be used to specify an optional path, such as www.example.com/project/help.htm. Additionally, you can specify which geographic locations you would like the probe to launch from by using the `vantagePointNames` field. For a list of available vantage points, see Vantage Points.

A 200 response will be returned with the successful creation of a probe, and the results of the probe can be retrieved from the URL in the `resultsUrl` field of the response. It will take a few moments for results to display once the tests have been configured.

```
{
   "id":"ocid1.pingprobe.OC2...abuxgljr6l4nepxjkmbtnibwqpu6y34xdvmr7okzoi47wicoflrxh32rwd9z",
   "compartmentId":"ocid1.compartment.oc1..aaaaaaacsxgoz7flzmry5dr6scf7xru6lwqpoygqld74npvn5rdj2cv3iiq",
   "resultsUrl":"https://healthchecks.us-ashburn-
```

```
1.oraclecloud.com/20180501/pingProbeResults/ocid1.pingprobe.OC2...abuxgljr6l4nepxjkmbtnibwqpu6y34xdvmr7
okzoi47wicoflrxh32rwd9z",
   "targets":[
      "www.example.com"
   ],
   "vantagePointNames":[
      "ibm-sjc",
      "aws-dub",
      "dgo-nyc"
   ],
   "protocol":"ICMP",
   "timeoutInSeconds":30
}
```

For more information about creating a probe, see **CreateOnDemandHttpProbe**.



Note

You can configure similar style probes using TCP or ICMP protocols. For more information, see **CreateOnDemandPingProbe**.

Retrieving Probe And Monitor Results

Probe and monitor results can be retrieved from URL in the `resultsUrl` field of a monitor or probe creation response. It will take a few moments for results to display once the tests have been configured. Results can also be retrieved at anytime using the following methods:

- **ListPingProbeResults** - For monitors or on-demand probes using TCP or ICMP protocols.
- **ListHttpProbeResults** - For monitors or on-demand probes using HTTP protocols.

Retrieving results for an on-demand probe or monitor requires the probe or monitor's configuration ID as a parameter. On-demand probe and configuration IDs are assigned upon their creation and are returned in the id field of the POST response. You can also use the **ListHttpMonitor** method to retrieve a list of currently configured monitors and probes using

HTTP protocols. Use the **ListPingMonitors** method to retrieve a list of currently configured monitors and probes using TCP and ICMP protocols.

The following is an example of results retrieved using `GET /httpProbeResults/{probeConfigurationId}`.


```
{
  "key": "651b9f3a46041cace0530204060ae27e",
  "probeConfigurationId":
"ocid1.httpmonitor.OC2...abuxgljr6l4nepxjkmbtnibwqpu5z24xdvmr7okzoi47wicoflrxh32rwd7a",
  "startTime": 1517323711505,
  "target": "www.example.com",
  "vantagePointName": "dgo-nyc",
  "protocol": "HTTPS",
  "connection": {
    "connectDuration": 114,
    "secureConnectDuration": 99,
    "address": "93.184.216.34",
    "port": 443
    },
  "dns":{
    "domainLookupDuration": 29,
    "addresses": [
      "93.184.216.34",
      "2606:2800:220:1:248:1893:25c8:1946"
      ]
      },
  "statusCode": 200,
  "fetchStart": 1517323711505,
  "domainLookupStart": 1517323711505,
  "domainLookupEnd": 1517323711534,
  "connectStart": 1517323711535,
  "secureConnectionStart": 1517323711550,
  "connectEnd": 1517323711649,
  "requestStart": 1517323711649,
  "responseStart": 1517323711673,
  "responseEnd": 1517323711676,
  "duration": 171,
  "encodedBodySize": 1270,
  "isTimedOut": false,
  "isHealthy": true
}
```

Vantage Points

Vantage points are geographic locations from which monitors and probes can be launched. Oracle Cloud Infrastructure maintains vantage points on the infrastructure of cloud providers around the world, including AWS, IBM, and Azure. The list below is a sampling of the vantage points available. The list of vantage points is dynamic and changes frequently. Use the ListHealthChecksVantagePoints method to return a list of available vantage points.

Provider	Location	Name
Amazon	Singapore	aws-sin
Amazon	Sao Paolo	aws-sao
Amazon	Dublin	aws-dub
Amazon	San Francisco	aws-sfo
Azure	Dublin	azr-dub
Azure	Amsterdam	azr-ams
Azure	Singapore	azr-sin
Azure	Sydney	azr-syd
Digital Ocean	Toronto	dgo-yyz
Digital Ocean	Frankfurt	dgo-fra
Digital Ocean	New York City	dgo-nyc
Digital Ocean	Biratnagar, Nepal	dgo-blr
Google	Taiwan	goo-tpe
Google	Brussels	goo-bru
Google	Council Bluffs, IA	goo-cbf

Provider	Location	Name
Google	Charleston, SC	goo-chs
IBM	San Jose, CA	ibm-sjc
IBM	Tokyo	ibm-hnd
IBM	Dallas, TX	ibm-dfw
IBM	Hong Kong	ibm-hkg
Rack Space	Ashburn, VA	rck-iad
Rack Space	Dallas, TX	rck-dfw
Rack Space	London	rck-lhr
Rack Space	Sydney	rck-syd

Managing Health Checks

The Health Checks service allows you to monitor the health of IP addresses and hostnames, as measured from geographic vantage points of your choosing, using HTTP and ping probes. After configuring a health check, you can view the monitor's results. The results include the location from which the host was monitored, the availability of the endpoint, and the date and time the test was performed.

Using the Console

MANAGING HEALTH CHECKS

To add a health check

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge**

Services and click **Health Checks**.

2. Click **Create Health Check**.
3. In the **Create Health Check** dialog box, enter the following:
    - **Health Check Name**: The name used for the health check. Avoid entering confidential information.
    - **Compartment**: Select the compartment the health check runs in.
    - **Target**: The IP address of the host being monitored. (Optional) Click **+ Add Target** to add multiple targets in succession.
    - **Vantage Point(s)**: Select the location from which the health of the target is monitored. No more than ten vantage points can be added.
    - **Type**: Select the type of request sent to monitor the target.
    - **Protocol**: The network protocol used to interact with your endpoint, such as HTTP protocol, which initializes an HTTP handshake with your endpoint.
    - **Port**: The port for the monitor to look for a connection. The default is port 80. For HTTPS, use port 8080.
    - **Path (Optional)**: The specific path on the target to be monitored.
    - **Header Name**: (Optional) The name displayed in the request header as part of the health check. Avoid entering confidential information.
    - **Header Value**: (Optional) Specifies the data requested by the header. Click **+ Add Header** to add multiple headers in succession.
    - **Method**: Select the HTTP method used for the health check.
    - **Interval**: Select the period of time between health checks of the target.
    - **Timeout**: Select the maximum time to wait for a reply before marking the health check as failed.
    - **Tags**: Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For

more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

4. Click **Create Health Check**.

The health check is added to the health check list. To view more details, click the health check name. It will take a few moments for results to display once the tests have been configured.

To edit a health check

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **Health Checks**.
2. Select the check box for the health check you want to edit.



Tip

To help find a health check, you can enter the name of the health check in the **Search** field.

3. Select **Edit** from the **Actions** drop-down menu.
4. In the Edit Health Check dialog box, make the needed changes, and then click **Edit Health Check**.

To disable a health check

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **Health Checks**.
2. Select the check box for the health check you want to disable.



Tip

To help find a health check, you can enter the name of the health check in the **Search** field.

3. Select **Disable** from the **Actions** drop-down menu.

The status of the health check changes to **Disabled** in the health check list.

To duplicate a health check

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **Health Checks**.
2. Select the check box for the health check you want to duplicate.
3. Select **Duplicate** from the **Actions** drop-down menu.
4. In the Create Health Check dialog box, make any updates to the duplicated health check, and then click **Create Health Check**.

To delete a health check

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **Health Checks**.
2. Select the check box for the health check you want to delete.



Tip

To help find a health check, you can enter the name of the health check in the **Search** field.

3. Select **Delete** from the **Actions** drop-down menu.

4. In the confirmation dialog box, click **Delete**.

To view the history of a health check

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **Health Checks**.
2. Click the name of the health check you want to view.

   The Health Check history displays a list of results for the past 90 days.

   Tip

   To help find a result, you can use the **Start Date**, **Start Time**, **End Date**, and **End Time** filter options.

3. Click the drop-down arrow beside the **TimeStamp** to view the monitor result details. You can use the API to download the data.

To manage tags for a health check

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **Health Checks**.
2. Click the name of the health check you want to view.
3. Click the **Tags** tab to view or edit the existing tags. Or click **Apply tag(s)** to add new ones.

For more information, see Resource Tags.



Tip

If your health checks are continually failing, please ensure that you have permission to monitor the host and that the ports on the host have been configured to receive traffic from Health Checks.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

- Use the CreateHTTPMonitor operation to create a Health Check monitor that uses the HTTP protocol.
- Use the CreatePingMonitor operation to create a Health Check monitor that uses the ping protocol.
- Use the ListHealthChecksVantagePoints to retrieve a list of available vantage points from which to execute monitors.
- Use the UpdateHttpMonitor operation to update the configuration of an HTTP health check monitor. You can also use this operation to disable an HTTP monitor by setting the `isEnabled` field to `false`.
- Use the UpdatePingMonitor operation to update the configuration of ping health check monitor. You can also use this operation to disable a ping monitor by setting the `isEnabled` field to `false`.
- Use the DeleteHttpMonitor operation to remove an HTTP health check monitor from your setup.
- Use the DeletePingMonitor operation to remove a ping health check monitor from your setup.

- Use the ListHttpProbeResults operation to retrieve the results of an HTTP health check monitor.
- Use the ListPingProbeResults operation to retrieve results of a ping health check monitor.

Overview of the Traffic Management Steering Policies Service

The Oracle Cloud Infrastructure Traffic Management Steering Policies service is a critical component of DNS. Traffic Management Steering Policies enables you to configure policies to serve intelligent responses to DNS queries, meaning different answers (endpoints) may be served for the query depending on the logic the customer defines in the policy. Traffic Management Steering Policies can account for health of answers to provide failover capabilities, provide the ability to load balance traffic across multiple resources, and account for the location where the query was initiated to provide a simple, flexible and powerful mechanism to efficiently steer DNS traffic.

Traffic Management Steering Policies Service Components

The following list describes the components used to build a traffic management steering policy.

STEERING POLICIES

A framework to define the traffic management behavior for your zones. Steering policies contain rules that help to intelligently serve DNS answers.

ATTACHMENTS

Allows you to link a steering policy to your zones. An attachment of a steering policy to a zone occludes all records at its domain that are of a covered record type, constructing DNS responses from its steering policy rather than from those domain's records. A domain can have at most one attachment covering any given record type.

RULES

The guidelines steering policies use to filter answers based on the properties of a DNS request, such as the requests geo-location or the health of your endpoints.

ANSWERS

Answers contain the DNS record data and metadata to be processed in a steering policy.

Ways to Access the Traffic Management Steering Policies Service

You can access Oracle Cloud Infrastructure using the Console (a browser-based interface) or the REST API. Instructions for the Console and API are included in topics throughout this guide.

To access the Console, you must use a supported browser. You can use the Console link at the top of this page to go to the sign-in page. Enter your tenancy, user name, and your password.

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

Traffic Management Steering Policies Service Capabilities and Limits

The Oracle Cloud Infrastructure Traffic Management Steering Policies service is limited to 100 policies and 1,000 attachments per tenant. See Service Limits for a list of applicable limits and instructions for requesting a limit increase.

Required IAM Service Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

If you're new to policies, see Getting Started with Policies and Common Policies. For more details about policies for Traffic Management Steering Policies, see Details for the Traffic Management Steering Policies Service.

Traffic Management Steering Policies API Guide

Traffic Management Steering Policies allows you to build and configure traffic management policies using the Oracle Cloud Infrastructure DNS REST API. Use the following guide to learn how policies are constructed using the REST API.

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

Traffic Management Steering Policy Components

The following list describes the components used to build a Traffic Management Steering Policy.

STEERING POLICIES

An overall framework to define the traffic management behavior for your zones. Steering policies contain rules that help to intelligently serve DNS answers.

ATTACHMENTS

Allows you to link a steering policy to your zones. An attachment of a steering policy to a zone occludes all records at its domain that are of a covered record type, constructing DNS responses from its steering policy rather than from those domain's records. A domain can have at most one attachment covering any given record type.

RULES

The guidelines steering policies use to filter answers based on the properties of a DNS request, such as the requests geo-location or the health of your endpoints.

ANSWERS

Answers contain the DNS record data and metadata to be processed in a steering policy.

TEMPLATES

Templates are predefined rule sequences that create a policy type and its intended behavior. Example: The `FAILOVER` template determines answers by checking DNS query against a `FILTER` rule first, then the following rules in succession: `HEALTH`, `PRIORITY`, and `LIMIT`. This gives the domain dynamic failover capability. Policies that define the `template` field with any policy other than `CUSTOM`, must follow the rule sequence outlined

for that policy type, otherwise, a `400` status code error will be returned upon policy creation.

CASES

A rule may optionally include a sequence of cases defining alternate configurations for how it should behave during processing for any given DNS query. When a rule has no sequence of cases, it is always evaluated with the same configuration during processing. When a rule has an empty sequence of cases, it is always ignored during processing. When a rule has a non-empty sequence of cases, its behavior during processing is configured by the first matching case in the sequence. A rule case with no `caseCondition` always matches. A rule case with a `caseCondition` matches only when that expression evaluates to true for the given query.

Create Steering Policies Using Templates

The following section explains the rule configuration for each type of steering policy template followed by an example POST request (CreateSteeringPolicy) displaying how to configure each template.

FAILOVER

Failover policies allow you to prioritize the order in which you want answers served in a policy (for example, Primary and Secondary). Oracle Cloud Infrastructure Health Checks are leveraged to determine the health of answers in the policy. If the Primary Answer is determined to be unhealthy, DNS traffic will automatically be steered to the Secondary Answer. Each of the following rules must be defined in the order specified below in the `rules` field of your request body when using a `FAILOVER` template:

Order	Rule	Restrictions	Comments
1	FILTER	• No cases are allowed. • Answer data must be defined in `defaultAnswerData` using the following JSON: `{ "answerCondition": "answer.isDisabled != true", "shouldKeep": true }`	
2	HEALTH	• No cases are allowed.	Only included if `healthCheckMonitorId` is defined for the policy.

Example of a POST /steeringPolicies policy using the `FAILOVER` template:

```
{
  "compartmentId": "ocid1...",
  "displayName": "failover between endpoints",
  "ttl": 30,
  "healthCheckMonitorId": "ocid1...",
  "template": "FAILOVER",
  "answers": [
    {
      "name": "server-primary",
      "rtype": "A",
      "rdata": "192.0.2.1",
      "pool": "primary"
    },
    {
      "name": "server-secondary",
      "rtype": "A",
      "rdata": "192.1.2.1",
      "pool": "secondary"
    }
  ],
```

```
  "rules": [
    {
      "ruleType": "FILTER",
      "defaultAnswerData": [
        {
          "answerCondition": "answer.isDisabled != true",
          "shouldKeep": true
        }
      ]
    },
    {
      "ruleType": "HEALTH"
    },
    {
      "ruleType": "PRIORITY",
      "defaultAnswerData": [
        {
          "answerCondition": "answer.pool == 'primary'",
          "value": 1
        },
        {
          "answerCondition": "answer.pool == 'secondary'",
          "value": 99
        }
      ]
    },
    {
      "ruleType": "LIMIT",
      "defaultCount": 1
    }
  ]
}
```

LOAD_BALANCE

Load Balancer policies allow distribution of traffic across multiple endpoints. Endpoints can be assigned equal weights to distribute traffic evenly across the endpoints or custom weights may be assigned for ratio load balancing. Oracle Cloud Infrastructure Health Checks are leveraged to determine the health of the endpoint. DNS traffic will be automatically distributed to the other endpoints, if an endpoint is determined to be

unhealthy. Each of the following rules must be defined in the order specified below in the `rules` field of your request body when using a `LOAD_BALANCE` template:

Order	Rule	Restrictions	Comments
1	`FILTER`	• No cases are allowed. • Answer data must be defined in `defaultAnswerData` using the following JSON: `{ "answerCondition": "answer.isDisabled != true", "shouldKeep": true }`	
2	`HEALTH`	• No cases are allowed.	Only included if `healthCheckMonitorId` is defined for the policy.

Example of a `POST /steeringPolicies` request body using the `LOAD_BALANCE` template:

```
{
  "compartmentId": "ocid1...",
  "displayName": "Weighted load balance for a set of answers with health checks",
  "ttl": 30,
  "healthCheckMonitorId": "ocid1...",
  "template": "LOAD_BALANCE",
  "answers": [
    {
      "name": "server1",
      "rtype": "A",
      "rdata": "192.0.2.1"
    },
    {
      "name": "server2",
      "rtype": "A",
      "rdata": "198.51.100.1"
    }
```

```
  ],
  "rules": [
    {
      "ruleType": "FILTER",
      "defaultAnswerData": [
        {
          "answerCondition": "answer.isDisabled != true",
          "shouldKeep": true
        }
      ]
    },
    {
      "ruleType": "HEALTH"
    },
    {
      "ruleType": "WEIGHTED",
      "defaultAnswerData": [
        {
          "answerCondition": "answer.name == 'server1'",
          "value": 99
        },
        {
          "answerCondition": "answer.name == 'server2'",
          "value": 1
        }
      ]
    },
    {
      "ruleType": "LIMIT",
      "defaultCount": 1
    }
  ]
}
```

ROUTE_BY_GEO

Geolocation-based steering policies distribute DNS traffic to different endpoints based on the location of the end user. Customers can define geographic regions composed of originating continent, countries or states/provinces (North America) and define a separate endpoint or set of endpoints for each region. Each of the following rules must be defined in

the order specified below in the `rules` field of your request body when using a `ROUTE_BY_GEO` template:

Order	Rule	Restrictions	Comments
1	`FILTER`	• No cases are allowed. • Answer data must be defined in `defaultAnswerData` using the following JSON: `{ "answerCondition": "answer.isDisabled != true", "shouldKeep": true }`	
2	`HEALTH`	• No cases are allowed.	Only included if `healthCheckMonitorId` is defined for the policy.

Example of a `POST /steeringPolicies` request body using the `ROUTE_BY_GEO` template:

```
{
  "compartmentId": "ocid1...",
  "displayName": "Geolocations mapped to answer pools",
  "ttl": 30,
  "healthCheckMonitorId": "ocid1...",
  "template": "ROUTE_BY_GEO",
  "answers": [
    {
      "name": "US Server 1",
      "rtype": "A",
      "rdata": "10.10.10.10",
      "pool": "US"
    },
    {
      "name": "US Server 2",
      "rtype": "A",
      "rdata": "10.10.10.11",
```

```
      "pool": "US"
    },
    {
      "name": "EU Server 1",
      "rtype": "A",
      "rdata": "10.10.1.1",
      "pool": "EU"
    },
    {
      "name": "EU Server 2",
      "rtype": "A",
      "rdata": "10.10.1.2",
      "pool": "EU"
    },
    {
      "name": "rest of world 1",
      "rtype": "A",
      "rdata": "203.0.113.1",
      "pool": "Global"
    },
    {
      "name": "rest of world 2",
      "rtype": "A",
      "rdata": "203.0.113.2",
      "pool": "Global"
    }
  ],
  "rules": [
    {
      "ruleType": "FILTER",
      "defaultAnswerData": [
        {
          "answerCondition": "answer.isDisabled != true",
          "shouldKeep": true
        }
      ]
    },
    {
      "ruleType": "HEALTH"
    },
```

```
{
  "ruleType": "PRIORITY",
  "cases": [
    {
      "caseCondition": "query.client.geoKey in (geoKey '6255149')",
      "answerData": [
        {
          "answerCondition": "answer.pool == 'US'",
          "value": 1
        },
        {
          "answerCondition": "answer.pool == 'EU'",
          "value": 2
        },
        {
          "answerCondition": "answer.pool == 'Global'",
          "value": 3
        }
      ]
    },
    {
      "caseCondition": "query.client.geokey in (geokey '6255148')",
      "answerdata": [
        {
          "answerCondition": "answer.pool == 'EU'",
          "value": 1
        },
        {
          "answerCondition": "answer.pool == 'US'",
          "value": 2
        },
        {
          "answerCondition": "answer.pool == 'Global'",
          "value": 3
        }
      ]
    },
    {
      "answerData": [
        {
```

```
            "answerCondition": "answer.pool == 'Global'",
            "value": 1
          },
          {
            "answerCondition": "answer.pool == 'US'",
            "value": 2
          },
          {
            "answerCondition": "answer.pool == 'EU'",
            "value": 3
          }
        ]
      }
    ]
  },
  {
    "ruleType": "LIMIT",
    "defaultCount": 1
  }
 ]
}
```

For a list of geographic keys to use in the `geokey` field, see Traffic Management Steering Policy geokeys.

ROUTE_BY_ASN

ASN-based steering policies enable you to steer DNS traffic based on Autonomous System Numbers (ASN). DNS queries originating from a specific ASN or set of ASNs can be steered to a specified endpoint. Each of the following rules must be defined in the order specified below in the `rules` field of your request body when using a `ROUTE_BY_ASN` template:

Order	Rule	Restrictions	Comments
1	FILTER	• No cases are allowed. • Answer data must be defined in `defaultAnswerData` using the following JSON: `{ "answerCondition": "answer.isDisabled != true", "shouldKeep": true }`	
2	HEALTH	• No cases are allowed.	Only included if `healthCheckMonitorId` is defined for the policy.

Example of a `POST /steeringPolicies` request body using the `ROUTE_BY_ASN` template:

```
{
  "compartmentId": "ocid1...",
  "displayName": "ASNs mapped to pools",
  "ttl": 30,
  "template": "ROUTE_BY_ASN",
  "answers": [
    {
      "name": "MIT Server",
      "rtype": "A",
      "rdata": "10.10.10.10",
      "pool": "MIT"
    },
    {
      "name": "Google Fiber Server",
      "rtype": "A",
      "rdata": "10.10.1.1",
      "pool": "Google Fiber"
    },
    {
      "name": "Other",
```

```
    "rtype": "A",
    "rdata": "203.0.113.1",
    "pool": "Other"
  }
],
"rules": [
  {
    "ruleType": "FILTER",
    "defaultAnswerData": [
      {
        "answerCondition": "answer.isDisabled != true",
        "shouldKeep": true
      }
    ]
  },
  {
    "ruleType": "PRIORITY",
    "cases": [
      {
        "caseCondition": "query.client.asn == 3",
        "answerData": [
          {
            "answerCondition": "answer.pool == 'MIT'",
            "value": 1
          },
          {
            "answerCondition": "answer.pool == 'Google Fiber'",
            "value": 2
          },
          {
            "answerCondition": "answer.pool == 'Other'",
            "value": 3
          }
        ]
      },
      {
        "caseCondition": "query.client.asn == 16591",
        "answerdata": [
          {
            "answerCondition": "answer.pool == 'Google Fiber'",
```

```
            "value": 1
          },
          {
            "answerCondition": "answer.pool == 'MIT'",
            "value": 2
          },
          {
            "answerCondition": "answer.pool == 'Other'",
            "value": 3
          }
        ]
      },
      {
        "answerData": [
          {
            "answerCondition": "answer.pool == 'Other'",
            "value": 1
          },
          {
            "answerCondition": "answer.pool == 'MIT'",
            "value": 2
          },
          {
            "answerCondition": "answer.pool == 'Google Fiber'",
            "value": 3
          }
        ]
      }
    ]
  },
  {
    "ruleType": "LIMIT",
    "defaultCount": 1
  }
 ]
}
```

ROUTE_BY_IP

IP Prefix-based steering policies enable customers to steer DNS traffic based on the IP Prefix of the originating query. Each of the following rules must be defined in the order

specified below in the `rules` field of your request body when using a `ROUTE_BY_IP` template:

Order	Rule	Restrictions	Comments
1	`FILTER`	• No cases are allowed. • Answer data must be defined in `defaultAnswerData` using the following JSON: `"answer.isDisabled != true",`	`{` `"answerCondition":` `"shouldKeep": true` `}`
2	`HEALTH`	• No cases are allowed.	Only included if `healthCheckMonitorId` is defined for the policy.

Example of a `POST /steeringPolicies` request body using the `ROUTE_BY_IP` template:

```
{
  "compartmentId": "ocid1...",
  "displayName": "IP subnets mapped to answer pools",
  "ttl": 30,
  "template": "ROUTE_BY_IP",
  "answers": [
    {
      "name": "MIT Server",
      "rtype": "A",
      "rdata": "10.10.10.10",
      "pool": "MIT"
    },
    {
      "name": "Google Fiber Server",
      "rtype": "A",
      "rdata": "10.10.1.1",
```

```
      "pool": "Google Fiber"
    },
    {
      "name": "Other",
      "rtype": "A",
      "rdata": "203.0.113.1",
      "pool": "Other"
    }
  ],
  "rules": [
    {
      "ruleType": "FILTER",
      "defaultAnswerData": [
        {
          "answerCondition": "answer.isDisabled != true",
          "shouldKeep": true
        }
      ]
    },
    {
      "ruleType": "PRIORITY",
      "cases": [
        {
          "caseCondition": "query.client.address in (subnet '18.0.0.0/9')",
          "answerData": [
            {
              "answerCondition": "answer.pool == 'MIT'",
              "value": 1
            },
            {
              "answerCondition": "answer.pool == 'Google Fiber'",
              "value": 2
            },
            {
              "answerCondition": "answer.pool == 'Other'",
              "value": 3
            }
          ]
        },
        {
```

```
        "caseCondition": "query.client.address in (subnet '136.32.0.0/11')",
        "answerdata": [
          {
            "answerCondition": "answer.pool == 'Google Fiber'",
            "value": 1
          },
          {
            "answerCondition": "answer.pool == 'MIT'",
            "value": 2
          },
          {
            "answerCondition": "answer.pool == 'Other'",
            "value": 3
          }
        ]
      },
      {
        "answerData": [
          {
            "answerCondition": "answer.pool == 'Other'",
            "value": 1
          },
          {
            "answerCondition": "answer.pool == 'MIT'",
            "value": 2
          },
          {
            "answerCondition": "answer.pool == 'Google Fiber'",
            "value": 3
          }
        ]
      }
    ]
  },
  {
    "ruleType": "LIMIT",
    "defaultCount": 1
  }
 ]
}
```

CUSTOM

Custom policies allow you to create complex policies combining the capabilities of failover, load balancing, geolocation, ASN and IP prefix steering. Custom templates to not require a regimented sequence of rules and it is recommended to contact Oracle Cloud Infrastructure support before creating a custom policy.

Rule Types

FILTER

Uses boolean data associated with answers, keeping answers only if the rule's `shouldKeep` value is `true`.

HEALTH

Utilizes Oracle Cloud Health Check monitors to determine the health of your endpoints and add and remove answers from your policy as needed. A health check monitor must be referenced in a health rule to have an effect on the policy. For more information about Health Checks, see Health Checks.

WEIGHTED

Uses a number between 0 and 255 used to determine how often an answer will be served in relation to other answers. Answers with higher values are more likely to be returned.

PRIORITY

Uses an integer associated with each answer to sort answers from lowest to highest value. Example: An answer with a priority value of 1 would be returned before an answer with a priority value of 10 in the list of answers. Answers that do not have a priority value assigned to them will be moved to the end of the list of answers.

LIMIT

Uses a count property to filter away all but the first answers in the list.

Managing Traffic Management Steering Policies

Policy Types

FAILOVER

Failover policies allow you to prioritize the order in which you want answers served in a policy (for example, Primary and Secondary). Oracle Cloud Infrastructure Health Checks are leveraged to determine the health of answers in the policy. If the Primary Answer is determined to be unhealthy, DNS traffic will automatically be steered to the Secondary Answer.

LOAD BALANCER

Load Balancer policies allow distribution of traffic across multiple endpoints. Endpoints can be assigned equal weights to distribute traffic evenly across the endpoints or custom weights may be assigned for ratio load balancing. Oracle Cloud Infrastructure Health Checks are leveraged to determine the health of the endpoint. DNS traffic will be automatically distributed to the other endpoints, if an endpoint is determined to be unhealthy.

GEOLOCATION STEERING

Geolocation steering policies distribute DNS traffic to different endpoints based on the location of the end user. Customers can define geographic regions composed of originating continent, countries or states/provinces (North America) and define a separate endpoint or set of endpoints for each region.

ASN STEERING

ASN steering policies enable you to steer DNS traffic based on Autonomous System Numbers (ASN). DNS queries originating from a specific ASN or set of ASNs can be steered to a specified endpoint.

IP PREFIX STEERING

IP Prefix steering policies enable customers to steer DNS traffic based on the IP Prefix of the originating query.

Typical Traffic Steering Scenarios

This section describes several typical scenarios for using Traffic Management Steering Policies.

Basic Failover

You can leverage Traffic Management Steering Policies to provide automated failover between primary and secondary servers.

Cloud Migration

Weighted load balancing supports controlled migration from your data center to Oracle Cloud Infrastructure servers. You can steer a small amount of traffic (1%) to your new resources in the cloud to verify everything is working as expected. You can then increase the ratios until you are comfortable with fully migrating all DNS traffic to the cloud.

Load Balancing Across Multiple Servers for Scale

You can configure load balancing pools of multiple servers. Traffic Management Steering Policies can automatically distribute DNS traffic across the set of servers. Health Checks may also be used and traffic will be automatically redirected to healthy servers, if a server is determined to be unhealthy.

Hybrid Environments

Since Traffic Management Steering Policies is an agnostic service, it may be used to not only steer traffic to Oracle Cloud Infrastructure resources, but can also be used to steer traffic to any publicly exposed (internet resolvable) resources, including other cloud providers and enterprise data centers.

Worldwide Geolocation Treatment

You can divide your global users into geographically defined regions (for example, state/province level in NA, country level for rest of world) and steer customers to specified resources based on their location. This helps to ensure global, high performing internet resolution, and supports functions such as ring fencing. For example, keeping traffic from China in China and block traffic outside of China into China.

CANARY TESTING

Leveraging IP Prefix steering, you can configure policies to serve different responses for your internal users versus external users.

ZERO-RATING SERVICES

ASN steering conditional steering based on the originating enterprise, mobile operator or other communications provider in support of various commercial agreements that may be in place. Essentially, preferred ASNs can be directed to free resources, while all other traffic can be directed to paid resources.

Using the Console

MANAGING TRAFFIC MANAGEMENT STEERING POLICIES

To create a Load Balancer policy

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **Traffic Management Steering Policies**.
2. Click **Create Traffic Management Steering Policy**.
3. In the **Create Traffic Management Steering Policy** dialog box, select **Load Balancer**.
4. Enter the following information:
    - **Policy Name:** The unique name that identifies policy.
    - **Policy TTL**: The Time to Live for responses from the steering policy. If not specified, the system will set this value on the steering policy.
    - **Maximum Answer Count**: The maximum number of answers returned for the policy.
    - **Answer(s):** Answer pools contain the group of answers that will be served in response to DNS queries.

- **Name**: A unique name to identify the answer. Avoid entering confidential information.
- **Type**: The record type that will be provided as the answer.
- **RDATA**: A valid domain name or IP address to add as an answer.
- **Weight**: A number between 0 and 255 used to determine how often an answer is served in relation to other answers. Answers with higher values are more likely to be served.
- **Eligible**: Select the check box to indicate that the answer is available within the pool to be used in response to queries. Alternatively, select **Mark pool answers eligible** or **Mark pool answers ineligible** from the **Actions** drop-down menu.

- **Attach Health Check:** Select an existing Health Check to be included as part of the policy, add a new one, or select **None**.
- **Attach Domain(s):** (Optional) The domain name and domain OCID you want to attach to the policy. Additional domains can be added in this section.

5. Click **Create Policy**.

The system creates and publishes the policy.

To create a Failover policy

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **Traffic Management Steering Policies**.
2. Click **Create Traffic Management Steering Policy**.
3. In the **Create Traffic Management Steering Policy** dialog box, select **Failover**.
4. Enter the following information:
   - **Policy Name:** The unique name that identifies policy. Avoid entering confidential information.

- **Policy TTL**: The Time to Live for responses from the steering policy. If not specified, the system will set this value on the steering policy.
- **Maximum Answer Count**: The maximum number of answers returned for the policy. For priority-based policies, the first valid answer is returned.
- **Answer Pool(s):** Answer pools contain the group of answers that will be served in response to DNS queries.
    - **Answer Pool Name**: A user-friendly name for the answer pool, unique within the steering policy. Avoid entering confidential information.
    - **Name**: A unique name to identify the answer. Avoid entering confidential information.
    - **Type**: The record type that will be provided as the answer.
    - **RDATA**: A valid domain name or IP address to add as an answer.
    - **Weight**: A number between 0 and 255 used to determine how often an answer is served in relation to other answers. Answers with higher values are more likely to be served.
    - **Eligible**: Select the check box to indicate that the answer is available within the pool to be used in response to queries. Alternatively, select **Mark pool answers eligible** or **Mark pool answers ineligible** from the **Actions** drop-down menu.
- **Pool Priority**: Failover priority rules specify the priority of answers that are served in a policy. If the primary answer is unavailable, traffic is steered to the next answer in the list.
    - **Pool**: Select the priority in which the answers are served.
- **Attach Health Check**: Select an existing Health Check to be included as part of the policy, add a new one, or select **None**.
- **Attach Domain(s)**: The domain name and domain OCID you want to attach to the policy. Additional domains can be added in this section.

5. Click **Create Policy**.

The system creates and publishes the policy.

To create a Geolocation Steering policy

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **Traffic Management Steering Policies**.
2. Click **Create Traffic Management Steering Policy**.
3. In the **Create Traffic Management Steering Policy** dialog box, select **Geolocation Steering**.
4. Enter the following information:
    - **Policy Name:** The unique name that identifies policy. Avoid entering confidential information.
    - **Policy TTL**: The Time to Live for responses from the steering policy. If not specified, the system will set this value on the steering policy.
    - **Maximum Answer Count**: The maximum number of answers returned for the policy. For priority-based policies, the first valid answer is returned.
    - **Answer Pool(s):** Answer pools contain the group of answers that will be served in response to DNS queries.
        - **Answer Pool Name**: A user-friendly name for the answer pool, unique within the steering policy. Avoid entering confidential information.
        - **Name**: A unique name to identify the answer. Avoid entering confidential information.
        - **Type**: The record type that will be provided as the answer.
        - **RDATA**: A valid domain name or IP address to add as an answer.
        - **Eligible**: Select the check box to indicate that the answer is available within the pool to be used in response to queries. Alternatively, select **Mark pool answers eligible** or **Mark pool answers ineligible** from the **Actions** drop-down menu.

- **Geolocation Steering Rules**: Geolocation steering rules specify the priority of answers that are served in a policy. If the primary answer is unavailable, traffic is steered to the next answer in the list. Additional rules and priorities can be added in this section.
  - **Geolocation**: Select a location that will be used to distribute DNS traffic.
  - **Pool Priority**: Select the priority in which the answers are served.
  - **Global Catch-all**: Adding a global catch-all allows you to specify answer pools for queries that do not match any of the specified rules you have added. Click **Add Global Catch-all** and select the pool priorities.
- **Attach Health Check**: Select an existing Health Check to be included as part of the policy, add a new one, or select **None**.
- **Attach Domain(s)**: The domain name and domain OCID you want to attach to the policy. Additional domains can be added in this section.

5. Click **Create Policy**.

The system creates and publishes the policy.

To create an ASN Steering policy

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **Traffic Management Steering Policies**.
2. Click **Create Traffic Management Steering Policy**.
3. In the **Create Traffic Management Steering Policy** dialog box, select **ASN Steering**.
4. Enter the following information:
   - **Policy Name:** The unique name that identifies policy. Avoid entering confidential information.
   - **Policy TTL**: The Time to Live for responses from the steering policy. If not specified, the system will set this value on the steering policy.

- **Maximum Answer Count**: The maximum number of answers returned for the policy. For priority-based policies, the first valid answer is returned.
- **Answer Pool(s):** Answer pools contain the group of answers that will be served in response to DNS queries.
    - **Answer Pool Name**: A user-friendly name for the answer pool, unique within the steering policy. Avoid entering confidential information.
    - **Name**: A unique name to identify the answer. Avoid entering confidential information.
    - **Type**: The record type that will be provided as the answer.
    - **RDATA**: A valid domain name or IP address to add as an answer.
    - **Eligible**: Select the check box to indicate that the answer is available within the pool to be used in response to queries. Alternatively, select **Mark pool answers eligible** or **Mark pool answers ineligible** from the **Actions** drop-down menu.
- **ASN Steering Rules**: ASN steering rules specify the priority of answers that are served in a policy. If the primary answer is unavailable, traffic is steered to the next answer in the list.
    - **ASN**: Enter an Autonomous System Number (ASN) that will be used to distribute DNS traffic.
    - **Pool Priority**: Select the priority in which the answers are served.
    - **Global Catch-all**: Adding a global catch-all allows you to specify answer pools for queries that do not match any of the specified rules you have added. Click **Add Global Catch-all** and select the pool priorities.
- **Attach Health Check**: Select an existing Health Check to be included as part of the policy, add a new one, or select **None**.
- **Attach Domain(s)**: The domain name and domain OCID you want to attach to the policy. Additional domains can be added in this section.

5. Click **Create Policy**.

The system creates and publishes the policy.

To create an IP Prefix Steering policy

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **Traffic Management Steering Policies**.
2. Click **Create Traffic Management Steering Policy**.
3. In the **Create Traffic Management Steering Policy** dialog box, select **IP Prefix Steering**.
4. Enter the following information:
    - **Policy Name:** The unique name that identifies policy. Avoid entering confidential information.
    - **Policy TTL**: The Time to Live for responses from the steering policy. If not specified, the system will set this value on the steering policy.
    - **Maximum Answer Count**: The maximum number of answers returned for the policy. For priority-based policies, the first valid answer is returned.
    - **Answer Pool(s):** Answer pools contain the group of answers that will be served in response to DNS queries.
        - **Answer Pool Name**: A user-friendly name for the answer pool, unique within the steering policy. Avoid entering confidential information.
        - **Name**: A unique name to identify the answer. Avoid entering confidential information.
        - **Type**: The record type that will be provided as the answer.
        - **RDATA**: A valid domain name or IP address to add as an answer.
        - **Eligible**: Select the check box to indicate that the answer is available within the pool to be used in response to queries. Alternatively, select **Mark pool answers eligible** or **Mark pool answers ineligible** from the **Actions** drop-down menu.

- **IP Prefix Steering Rules**: IP prefix steering rules specify the priority of answers that are served in a policy. If the primary answer is unavailable, traffic is steered to the next answer in the list.
    - **Subnet Address**: Enter a subnet address that will be used to distribute DNS traffic.
    - **Pool Priority**: Select the priority in which the answers are served.
    - **Global Catch-all**: Adding a global catch-all allows you to specify answer pools for queries that do not match any of the specified rules you have added. Click **Add Global Catch-all** and select the pool priorities.
- **Attach Health Check**: Select an existing Health Check to be included as part of the policy, add a new one, or select **None**.
- **Attach Domain(s)**: The domain name and domain OCID you want to attach to the policy. Additional domains can be added in this section.

5. Click **Create Policy**.

The system creates and publishes the policy.

To update a policy

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **Traffic Management Steering Policies**.
2. Click the **Policy Name** you want to update. Policy information and a list of attached domains appear.



Tip

You can use search for a policy by name in the **Search** field. You can also use the **Time Created** sort filter to sort the policies chronologically in ascending or descending order.

3. Click **Edit**.
4. Make the needed changes, and then click **Save**.

To attach a domain to an existing policy

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **Traffic Management Steering Policies**.
2. Click the **Policy Name** you want to update. Policy information and a list of attached domains appear.



Tip

You can use search for a policy by name in the **Search** field. You can also use the **Time Created** sort filter to sort the policies chronologically in ascending or descending order.

3. Click **Add Attached Domain(s)**.
4. In the Add Attached Domain(s) dialog box, enter the domain and select a zone.
5. Click **Submit**.

To edit an attached domain

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **Traffic Management Steering Policies**.
2. Click the **Policy Name** you want to update. Policy information and a list of attached domains appear.



Tip

You can use search for a policy by name in the **Search** field. You can also use the **Time Created** sort filter to sort the policies chronologically in ascending or descending order.

3. For the attached domain you want to edit, click the Actions icon (three dots), and then click **Edit Attached Domain**.
4. In the Attached Domain(s) dialog box, enter the domain and select a zone.
5. Click **Save**.

To delete a policy

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **Traffic Management Steering Policies**.
2. Select the check box for the policy you want to delete.
3. Click **Delete**. The policy is staged for deletion.
4. Click **Publish Changes** to delete the policy.
5. In the confirmation dialog box, click **Publish Changes**.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

For more information about managing steering policies using the API, see Traffic Management Steering Policies API Guide.

Use the following operations to manage your steering policies:

- CreateSteeringPolicy
- ListSteeringPolicies
- GetSteeringPolicy
- UpdateSteeringPolicy
- DeleteSteeringPolicy

Use the following operations to manage your steering policy attachments:

- CreateSteeringPolicyAttachment
- ListSteeringPolicyAttachments
- GetSteeringPolicyAttachment
- UpdateSteeringPolicyAttachment
- DeleteSteeringPolicyAttachment

Traffic Management Steering Policy geokeys

Use these keys as values for the `geokey` fields of `caseConditions` in `ROUTE_BY_GEO` steering policies.

Continent geokeys

Continent Name	geoKey
Africa	6255146
Antarctica	6255152
Asia	6255147
Europe	6255148
North America	6255149

Continent Name	geoKey
Oceania	6255151
South America	6255150

Country geokeys

Country Name	geoKey
Afghanistan(AF)	1149361
Aland Islands(AX)	661882
Albania(AL)	783754
Algeria(DZ)	2589581
American Samoa(AS)	5880801
Andorra(AD)	3041565
Angola(AO)	3351879
Anguilla(AI)	3573511
Antarctica(AQ)	6697173
Antigua and Barbuda(AG)	3576396
Argentina(AR)	3865483
Armenia(AM)	174982
Aruba(AW)	3577279
Australia(AU)	2077456

Country Name	geoKey
Austria(AT)	2782113
Azerbaijan(AZ)	587116
Bahamas(BS)	3572887
Bahrain(BH)	290291
Bangladesh(BD)	1210997
Barbados(BB)	3374084
Belarus(BY)	630336
Belgium(BE)	2802361
Belize(BZ)	3582678
Benin(BJ)	2395170
Bermuda(BM)	3573345
Bhutan(BT)	1252634
Bolivia(BO)	3923057
Bonaire, Saint Eustatius and Saba(BQ)	7626844
Bosnia and Herzegovina(BA)	3277605
Botswana(BW)	933860
Bouvet Island(BV)	3371123
Brazil(BR)	3469034
British Indian Ocean Territory(IO)	1282588

Country Name	geoKey
British Virgin Islands(VG)	3577718
Brunei(BN)	1820814
Bulgaria(BG)	732800
Burkina Faso(BF)	2361809
Burundi(BI)	433561
Cambodia(KH)	1831722
Cameroon(CM)	2233387
Canada(CA)	6251999
Cape Verde(CV)	3374766
Cayman Islands(KY)	3580718
Central African Republic(CF)	239880
Chad(TD)	2434508
Chile(CL)	3895114
China(CN)	1814991
Christmas Island(CX)	2078138
Cocos (Keeling) Islands(CC)	1547376
Colombia(CO)	3686110
Comoros(KM)	921929
Congo(CG)	2260494

Country Name	geoKey
Cook Islands(CK)	1899402
Costa Rica(CR)	3624060
Croatia(HR)	3202326
Cuba(CU)	3562981
Curacao(CW)	7626836
Cyprus(CY)	146669
Czech Republic(CZ)	3077311
Democratic Republic of the Congo(CD)	203312
Denmark(DK)	2623032
Djibouti(DJ)	223816
Dominica(DM)	3575830
Dominican Republic(DO)	3508796
East Timor(TL)	1966436
Ecuador(EC)	3658394
Egypt(EG)	357994
El Salvador(SV)	3585968
Equatorial Guinea(GQ)	2309096
Eritrea(ER)	338010
Estonia(EE)	453733

Country Name	geoKey
Ethiopia(ET)	337996
Falkland Islands(FK)	3474414
Faroe Islands(FO)	2622320
Fiji(FJ)	2205218
Finland(FI)	660013
France(FR)	3017382
French Guiana(GF)	3381670
French Polynesia(PF)	4030656
French Southern Territories(TF)	1546748
Gabon(GA)	2400553
Gambia(GM)	2413451
Georgia(GE)	614540
Germany(DE)	2921044
Ghana(GH)	2300660
Gibraltar(GI)	2411586
Greece(GR)	390903
Greenland(GL)	3425505
Grenada(GD)	3580239
Guadeloupe(GP)	3579143

Country Name	geoKey
Guam(GU)	4043988
Guatemala(GT)	3595528
Guernsey(GG)	3042362
Guinea(GN)	2420477
Guinea-Bissau(GW)	2372248
Guyana(GY)	3378535
Haiti(HT)	3723988
Heard Island and McDonald Islands(HM)	1547314
Honduras(HN)	3608932
Hong Kong(HK)	1819730
Hungary(HU)	719819
Iceland(IS)	2629691
India(IN)	1269750
Indonesia(ID)	1643084
Iran(IR)	130758
Iraq(IQ)	99237
Ireland(IE)	2963597
Isle of Man(IM)	3042225
Israel(IL)	294640

Country Name	geoKey
Italy(IT)	3175395
Ivory Coast(CI)	2287781
Jamaica(JM)	3489940
Japan(JP)	1861060
Jersey(JE)	3042142
Jordan(JO)	248816
Kazakhstan(KZ)	1522867
Kenya(KE)	192950
Kiribati(KI)	4030945
Kuwait(KW)	285570
Kyrgyzstan(KG)	1527747
Laos(LA)	1655842
Latvia(LV)	458258
Lebanon(LB)	272103
Lesotho(LS)	932692
Liberia(LR)	2275384
Libya(LY)	2215636
Liechtenstein(LI)	3042058
Lithuania(LT)	597427

Country Name	geoKey
Luxembourg(LU)	2960313
Macau(MO)	1821275
Macedonia(MK)	718075
Madagascar(MG)	1062947
Malawi(MW)	927384
Malaysia(MY)	1733045
Maldives(MV)	1282028
Mali(ML)	2453866
Malta(MT)	2562770
Marshall Islands(MH)	2080185
Martinique(MQ)	3570311
Mauritania(MR)	2378080
Mauritius(MU)	934292
Mayotte(YT)	1024031
Mexico(MX)	3996063
Micronesia(FM)	2081918
Moldova(MD)	617790
Monaco(MC)	2993457
Mongolia(MN)	2029969

Country Name	geoKey
Montenegro(ME)	3194884
Montserrat(MS)	3578097
Morocco(MA)	2542007
Mozambique(MZ)	1036973
Myanmar(MM)	1327865
Namibia(NA)	3355338
Nauru(NR)	2110425
Nepal(NP)	1282988
Netherlands(NL)	2750405
New Caledonia(NC)	2139685
New Zealand(NZ)	2186224
Nicaragua(NI)	3617476
Niger(NE)	2440476
Nigeria(NG)	2328926
Niue(NU)	4036232
Norfolk Island(NF)	2155115
North Korea(KP)	1873107
Northern Mariana Islands(MP)	4041468
Norway(NO)	3144096

Country Name	geoKey
Oman(OM)	286963
Pakistan(PK)	1168579
Palau(PW)	1559582
Palestinian territories(PS)	6254930
Panama(PA)	3703430
Papua New Guinea(PG)	2088628
Paraguay(PY)	3437598
Peru(PE)	3932488
Philippines(PH)	1694008
Pitcairn(PN)	4030699
Poland(PL)	798544
Portugal(PT)	2264397
Puerto Rico(PR)	4566966
Qatar(QA)	289688
Reunion(RE)	935317
Romania(RO)	798549
Russia(RU)	2017370
Rwanda(RW)	49518
Saint Barthelemy(BL)	3578476

Country Name	geoKey
Saint Helena(SH)	3370751
Saint Kitts and Nevis(KN)	3575174
Saint Lucia(LC)	3576468
Saint Martin(MF)	3578421
Saint Pierre and Miquelon(PM)	3424932
Saint Vincent and the Grenadines(VC)	3577815
Samoa(WS)	4034894
San Marino(SM)	3168068
Sao Tome and Principe(ST)	2410758
Saudi Arabia(SA)	102358
Senegal(SN)	2245662
Serbia(RS)	6290252
Seychelles(SC)	241170
Sierra Leone(SL)	2403846
Singapore(SG)	1880251
Sint Maarten(SX)	7609695
Slovakia(SK)	3057568
Slovenia(SI)	3190538
Solomon Islands(SB)	2103350

Country Name	geoKey
Somalia(SO)	51537
South Africa(ZA)	953987
South Georgia and the South Sandwich Islands(GS)	3474415
South Korea(KR)	1835841
South Sudan(SS)	7909807
Spain(ES)	2510769
Sri Lanka(LK)	1227603
Sudan(SD)	366755
Suriname(SR)	3382998
Svalbard and Jan Mayen(SJ)	607072
Swaziland(SZ)	934841
Sweden(SE)	2661886
Switzerland(CH)	2658434
Syria(SY)	163843
Taiwan(TW)	1668284
Tajikistan(TJ)	1220409
Tanzania(TZ)	149590
Thailand(TH)	1605651

Country Name	geoKey
Togo(TG)	2363686
Tokelau(TK)	4031074
Tonga(TO)	4032283
Trinidad and Tobago(TT)	3573591
Tunisia(TN)	2464461
Turkey(TR)	298795
Turkmenistan(TM)	1218197
Turks and Caicos Islands(TC)	3576916
Tuvalu(TV)	2110297
U.S. Virgin Islands(VI)	4796775
Uganda(UG)	226074
Ukraine(UA)	690791
United Arab Emirates(AE)	290557
United Kingdom(GB)	2635167
United States(US)	6252001
United States Minor Outlying Islands(UM)	5854968
Uruguay(UY)	3439705
Uzbekistan(UZ)	1512440
Vanuatu(VU)	2134431

Country Name	geoKey
Vatican City(VA)	3164670
Venezuela(VE)	3625428
Vietnam(VN)	1562822
Wallis and Futuna(WF)	4034749
Western Sahara(EH)	2461445
Yemen(YE)	69543
Zambia(ZM)	895949
Zimbabwe(ZW)	878675

United States geokeys

State Name	geoKey
Alabama	4829764
Alaska	5879092
Arizona	5551752
Arkansas	4099753
California	5332921
Colorado	5417618
Connecticut	4831725
Delaware	4142224

State Name	geoKey
District of Columbia	4138106
Florida	4155751
Georgia	4197000
Hawaii	5855797
Idaho	5596512
Illinois	4896861
Indiana	4921868
Iowa	4862182
Kansas	4273857
Kentucky	6254925
Louisiana	4331987
Maine	4971068
Maryland	4361885
Massachusetts	6254926
Michigan	5001836
Minnesota	5037779
Mississippi	4436296
Missouri	4398678
Montana	5667009

State Name	geoKey
Nebraska	5073708
Nevada	5509151
New Hampshire	5090174
New Jersey	5101760
New Mexico	5481136
New York	5128638
North Carolina	4482348
North Dakota	5690763
Ohio	5165418
Oklahoma	4544379
Oregon	5744337
Pennsylvania	6254927
Rhode Island	5224323
South Carolina	4597040
South Dakota	5769223
Tennessee	4662168
Texas	4736286
Utah	5549030
Vermont	5242283

State Name	geoKey
Virginia	6254928
Washington	5815135
West Virginia	4826850
Wisconsin	5279468
Wyoming	5843591

Canada Provinces geokeys

Province Name	geoKey
Alberta	5883102
British Columbia	5909050
Manitoba	6065171
New Brunswick	6087430
Newfoundland and Labrador	6354959
Northwest Territories	6091069
Nova Scotia	6091530
Nunavut	6091732
Ontario	6093943
Prince Edward Island	6113358
Quebec	6115047

Province Name	geoKey
Saskatchewan	6141242
Yukon	6185811

Overview of the Web Application Firewall Service

Oracle Cloud Infrastructure Web Application Firewall (WAF) is a cloud-based, Payment Card Industry (PCI) compliant, global security service that protects applications from malicious and unwanted internet traffic. WAF can protect any internet facing endpoint, providing consist rule enforcement across a customer's applications.

WAF provides you with the ability to create and manage rules for internet threats including Cross-Site Scripting (XSS), SQL Injection and other OWASP-defined vulnerabilities. Unwanted bots can be mitigated while tactically allowed desirable bots to enter. Access rules can limit based on geography or the signature of the request.

The global Security Operations Center (SOC) will continually monitor the internet threat landscape acting as an extension of your IT infrastructure.

Web Application Firewall Service Components

WEB APPLICATION FIREWALL POLICY

WAF policies encompass the overall configuration of your WAF service, including origin management, protection rule settings, and bot detection features.

ORIGIN

Your web application's origin host server. An origin must be defined in your WAF policy in order to set up protection rules or other features.

PROTECTION RULES

Protection rules can be configured to either allow, block, or log network requests when they meet the specified criteria of a protection rule. The WAF will observe traffic to your web

application over time and suggest new rules to apply. To view a list of available WAF rules, see Supported Protection Rules.

BOT MANAGEMENT

The WAF service includes several features that allow you to detect and either block or allow identified bot traffic to your web applications. Bot management features include: JavaScript Challenge, CAPTCHA Challenge, and GoodBot whitelists. For more information, see Bot Management.

Ways to Access the WAF Service

You can access Oracle Cloud Infrastructure using the Console (a browser-based interface), command line interface (CLI), or the REST API. Instructions for the Console and API are included in topics throughout this guide.

To access the Console, you must use a supported browser. You can use the Console link at the top of this page to go to the sign-in page. Enter your tenancy, user name, and your password.

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

Note About The API

The WAF service is powered by the Oracle Cloud Infrastructure Web Application Acceleration and Security (WAAS) API. All WAF related calls must be made using the WAAS API. To create a WAF configuration using the API, you must first create a WAAS policy with a defined origin and domain using the API. For the purposes of access control, you must provide the OCID of the compartment where you want the service to reside. For information about access control and compartments, see Overview of the IAM Service.

WAF Service Capabilities and Limits

The WAF service is limited to 50 policies per tenant. See Service Limits for instructions on requesting a limit increase.

Required IAM Service Policy

To use Oracle Cloud Infrastructure, you must be given access in a policy for waas-policy. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

Policy examples:

- To allow a specific user group to manage policies in the WAF:

```
Allow group <GroupName> to manage waas-policy in compartment <CompartmentName>
```

```
Allow group <GroupName> to read waas-work-request in compartment <CompartmentName>
```

- To allow a specific user group to manage certificates in the WAF:

```
Allow group <GroupName> to manage waas-certificate in compartment <CompartmentName>
```

- To allow a specific user group view policies in the WAF

```
Allow group <GroupName> to read waas-policy in tenancy <TenancyName>
```

If you're new to policies, see Getting Started with Policies and Common Policies. For more details about policies for WAF, see Details for the WAF Service.

Tagging Resources

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

Getting Started with WAF

If you're new to Oracle Cloud Infrastructure WAF, this topic gives guidance on how to proceed.

Before You Begin

To begin using the WAF service, you must have the following available if you plan to run your site on HTTPS/443:

- Public certificate for the fully qualified domain name (FQDN) of the application.
- Corresponding private key for the site.
- IP address of the LBaaS or other public facing endpoint of application.
- Ability to update DNS records for the domain.

Securing Your WAF

To secure your WAF, you must configure your servers to accept traffic from the WAF servers. Configure your origin's ingress rules to only accept connections from the following CIDR ranges:

- 192.157.18.0/23
- 205.147.88.0/21
- 192.69.118.0/23
- 198.181.48.0/21
- 199.195.6.0/23

Create a Policy to Route Traffic Through the WAF

To begin, create a policy to route traffic through the WAF without rules enabled. Creating a policy without rules enabled ensures that there are no regressions by having a reverse proxy in front of the application.

To create a policy

1. Select the region and compartment where the policy should be maintained (there is no constraint around the WAF co-existing with Load Balancing or other application resources in Oracle Cloud Infrastructure.
2. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
3. Click **Create WAF Policy**.
4. In the **Create WAF Policy** dialog box, enter the following:
    - **Policy Name:** A unique name for the policy. Avoid entering confidential information.
    - **Domains:**
        - **Primary Domain**: The fully qualified domain name (FQDN) of the application where the policy will be applied.
        - **Additional Domains**: (Optional) Subdomains where the policy will be applied.
    - **WAF Origin:** The host or IP address of the public internet facing application that is being protected by the application.
        - **Origin Name**: A unique name for the origin.
        - **URI** - Enter the public facing endpoint (IPv4 or FQDN) of the application.
        - **HTTPS Port**: The port used for secure HTTP connection. The default port is 443.
        - **HTTP Port**: The HTTP port the origin listens on. The default port is 80.

- **Header(s):** (Optional)
  - **Header Name**: The name displayed in the HTTP request header and the header value that can be added and passed to the origin server with all requests.
  - **Header Value**: Specifies the data requested by the header.

- **Tags**: Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

5. Click **Create WAF Policy**. The WAF Policy overview appears. Expect the policy to become active within 15 minutes of creation.

Update DNS to Enable WAF

In this step, you update the CNAME for your zone to route requests from internet clients to WAF. Use the following instructions to make this DNS change in the Console. If your DNS setup resides with another provider, refer to their documentation for instructions.

To update the CNAME for your zone

1. In the Policy Information tab of the WAF Policy overview, select the CNAME Target.
2. Copy the CNAME target to your clipboard.
3. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **DNS Zone Management**.
4. Click the Zone Name of the primary domain where you want to update the record. Zone details and a list of records appear.
5. Select the check box for the CNAME record and select **Edit** from the **Actions** drop-down menu.

6. In the **Edit Record** dialog box, update the Target field with the CNAME Target from your clipboard.
7. Click **Submit**.
8. Click **Publish Changes**.
9. In the confirmation dialog box, click **Publish Changes**.

Upload Your Certificate and Key

This step assumes that your site runs on HTTPS/443.

To upload your certificate and key

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of your WAF Policy. The WAF Policy overview appears.
3. Click **Settings**.
4. Click **Edit**.
5. In the **Edit Settings** dialog box, enter the following:
   - **WAF Origin:** Select the name and IP address of the origin.
   - **Enable HTTPS Support:** Select this option so that communications between the browser and web app are encrypted. Enter the following information:
     - **SSL Certificate:** Drag and drop, select, or paste a valid SSL certificate in PEM format. You must also include intermediate certificates (the website certificate must be first). The following is an example:

```
-----BEGIN CERTIFICATE-----
                                                                    <Base64_encoded_certi
                                                                    -----END CERTIFICATE-
                                                                    -----BEGIN CERTIFICAT
                                                                    <Intermediate_Base64_
                                                          -----END CERTIFICATE-----
```

- **Private Key:** Drag and drop, select, or paste a valid private key in PEM format in this field. The private key cannot be protected by a passphrase. The following is an example:

```
-----BEGIN PRIVATE KEY-----
                                                                    <Base64_encoded_priva
                                                  -----END PRIVATE KEY-----
```

- **Self Signed Certificate:** Enable this field when using a self-signed certificate to show an SSL warning in the browser.
- **HTTP to HTTPS Redirect:** When enabled, all HTTP traffic is automatically redirected to HTTPS.

6. Click **Save**. Updates to your WAF policy appear in the list to be published in Unpublished Changes.
7. In the WAF Policy overview, under **Unpublished Changes**, click **View**.
8. In the Unpublished Changes list, click the drop-down arrow beside an unpublished change to review the change.
9. Click **Publish All**.
10. In the Publish Changes dialog box, click **Publish All**.

Test Your Application

In this step, you ensure that requests are being routed to the WAF and that your application continues to function normally with a reverse proxy in the topology.

To test your application

1. Open a browser.
2. Enter the FQDN of the website protected by WAF.
3. Test the functionality of the application.
4. Inspect HTTP Response Headers to see if traffic is flowing through WAF. Some HTTP Response Headers to look for are:

- X-Cdn: Served-By-Zenedge
- Server: ZENEDGE

5. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
6. Click the name of your WAF Policy. The WAF Policy overview appears.
7. Under **Logs**, click **View**. Logs for the WAF policy appear.



Note

You may experience a one-minute delay on logs aggregated and available through the console.

8. Copy the IP address and User-Agent value of your requests to your clipboard. You can use this information when you enable WAF to passively detect for access rules.

Enable WAF to Passively Detect Rules

In this step, you enable WAF to detect protection rules without blocking requests. Enabling WAF to passively detect rules helps you visualize the traffic that may pose a threat to your site and help you tune the WAF to exclude false positives.

To enable WAF to detect protection rules

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to configure rule settings for. The WAF Policy overview appears.
3. Click **Protection Rules**.
4. Use the Protection Rules table to locate the rules you want to detect.

5. Enter the Rule IDs you located from the table into the **Rule ID** filter. For this example, enter **941140** (Cross-Site Scripting) in the **Rule ID** filter.
6. Select **Detect** from the **Actions** drop-down menu for the protection rules you filtered.

To enable WAF to detect access rules

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to configure access rules for. The WAF Policy overview appears.
3. Click **Access Control**.
4. Click **Add Access Rule**.
5. In the Add Access Rule dialog box, enter the following:
    a. **Name:** DetectRequestsFromMySpecificBrowser
    b. **Rule Condition:** Select **IP Address** from the drop-down and enter the IP address you copied to your clipboard while testing your application in the **IP Address** field.
    c. Click **+Additional Condition**.
    d. **Rule Condition:** Select **User Agent is** from the drop-down and enter the agent value you copied to your clipboard while testing your application in the **User Agent Header** field.
    e. **Rule Action:** Select **Detect Only**.



Note

Both the IP Address and the User Agent in the preceding example must match for the rule to be triggered. If a different User Agent is used to test your application, the request will not be detected.

6. Click **Add Access Rule**.
7. Click **Unpublished Changes**.
8. Click **Publish All**.

Test the Rules

When the policy is active, you can test that your rules are detected by WAF.

To initiate requests

1. Use the same browser you used when you tested your application to do the following:
   a. Request the FQDN of your application.
   b. Request the FQDN of your application with the following query parameter appended: `?id=<script>alert("TEST");</script>`.
2. Use a different browser on the same machine and repeat the preceding requests. All requests should go through the application.

To verify that WAF is detecting requests

To verify that WAF is detecting requests identified as a risk:

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.

2. Click the name of the WAF Policy you want to view logs for. The WAF Policy overview appears.
3. Click **Logs**. Logs for the WAF policy appear.
4. Select the **Detect** check box from the **Actions** filter.
5. Verify that there are two entries for the protection rule triggered by the Cross-Site Scripting request and one entry for detecting the User Agent and IP Address.

View Recommendations

To view protection rule recommendations

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to view protection rule recommendations for. The WAF Policy overview appears.
3. Click **Protection Rules**.
4. Click the **Recommendations** tab. This list is generated based on the traffic the WAF detects flowing through the WAF. If nothing appears in this list, keep testing the FQDN of your application and check back later.
5. Select the protection rules with a **Detect** recommended action and then click **Accept Recommendations**.



Tip

You can use the **Recommended Action** filter to locate a recommendation by **Detect**.

Enable WAF to Actively Block Requests

After you verify that requests are being detected, you can start blocking the undesired traffic.

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to configure rule settings for. The WAF Policy overview appears.
3. Click **Protection Rules**.
4. Enter rule ID **941140** in the **Rule ID** filter.
5. (Optional) To search for rules with a Detect action, select the **Detect** check box from the **Rule Action** filter.
6. Select **Block** from the **Actions** drop-down menu for rule ID **941140** and any other protection rules you filtered.
7. Under WAF Policy, click **Unpublished Changes**.
8. Click **Publish All**.
9. In the Publish Changes dialog box, click **Publish All**.
10. Test the rules again by initiating requests. You should get 403 Forbidden errors when testing with the JavaScript on the URL.

Managing WAF Policies

The Oracle Cloud Infrastructure WAF service enables you to create a WAF policy and origin.

Order of Processing

The order in which rules and handlers are processed is:

1. IP Whitelists/Blacklists/Good Bot Whitelists
2. Access Rules
3. JavaScript Challenge
4. Device Fingerprinting Challenge (available in the API)
5. Human Interaction Challenge (available in the API)
6. Captcha Challenge

7. Protection Rules
8. Rate Limiting (available in the API)

Using the Console

CREATE AND MANAGE WAF POLICIES

To create a WAF policy

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click **Create WAF Policy**.

3. In the **Create WAF Policy** dialog box, enter the following:
    - **Policy Name:** A unique name for the policy. Avoid entering confidential information.
    - **Domains:**
        - **Primary Domain**:The fully qualified domain name (FQDN) of the application where the policy will be applied.
        - **Additional Domains**: (Optional) Subdomains where the policy will be applied.
    - **WAF Origin:** The host or IP address of the public internet facing application that is being protected by the application.
        - **Origin Name**: A unique name for the origin. Avoid entering confidential information.
        - **URI**: The IPv4 address or fully qualified domain name (FQDN) of the origin. The URI can be a full URI, not just a host/IP.
        - **HTTPS Port**: The port used for secure HTTP connection. The default port is 443.
        - **HTTP Port**: The HTTP port the origin listens on. The default port is 80.
        - **Header(s):** (Optional)
            - **Header Name**: The name displayed in the HTTP request header and the header value that can be added and passed to the origin server with all requests.
            - **Header Value**: Specifies the data requested by the header.
    - **Tags**: Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

4. Click **Create WAF Policy**. The WAF Policy overview appears. You can access Origin Management, Access Control, WAF, Bot Management, Alerts, and any unpublished changes. While the policy is being created, no changes can be made until the process has completed. Expect the policy to become active within 15 minutes of creation.

   A CNAME target is generated for each policy. The CNAME target is a hyphenated version of your FQDN within the Oracle Cloud Infrastructure domain (for example, myapp-mydomain-com.oraclecloud.net).

5. In your DNS zone, update the CNAME record entry with the value of the CNAME target that is generated. This enables traffic to be routed through the WAF before the application. This value is presented soon after you publish your policy the first time on the main page of the policy.

To update a WAF policy

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to update. The WAF Policy overview appears.



Tip

You can use the Date Created sort filter to sort policies by the date they were created in ascending or descending order.

3. Click **Edit**.
4. In the Edit WAF Policy dialog box, make the needed changes and then click **Save**.

To delete a WAF policy

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Select the check box for the policy you want to delete.

> **Tip**
>
> You can use the Date Created sort filter to sort policies by the date they were created in ascending or descending order.

3. Click **Delete**.
4. In the confirmation dialog box, click **Delete**.

   The status of the policy changes from **Active** to **Deleting**. Deleted policies are maintained for a short time before they are unavailable in the Console.

To publish changes

Updates to your WAF policy appear in the list to be published in Unpublished Changes. Pending changes do not persist across browser sessions. Once you publish changes, it cannot be edited until changes propagate to the edge nodes.

1. In the WAF Policy overview, click **Unpublished Changes**.
2. In the Unpublished Changes list, click the drop-down arrow beside an unpublished change to review the change.
3. Click **Publish All**.
4. In the Publish Changes dialog box, click **Publish All**.

To manage tags for a WAF policy

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to view. The WAF Policy overview appears.
3. Click the **Tags** tab to view or edit existing tags. Or click **Apply tag(s)** to add new ones.

For more information, see Resource Tags.

Using the CLI

Open a command prompt and run the following command to get the details of a WAAS policy:

```
oci waas waas-policy get --waas-policy-id <policy_ocid>
```

This can be useful in retrieving the necessary information when opening a ticket with Oracle Cloud Infrastructure support. For more information about how to access and use the CLI, see Command Line Interface (CLI).

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

- CreateWaasPolicy
- GetWaasPolicy
- UpdateWaasPolicy
- DeleteWaasPolicy

Origin Management

An origin is an endpoint (typically an IP address) of the application protected by the WAF. An origin can be an Oracle Cloud Infrastructure load balancer public IP address. A load balancer IP address can be used for high availability to an origin. Multiple origins can be defined, but only a single origin can be active for a WAF.



Note

WAF supports default ports (80 and 443) on origin servers.

You can set HTTP headers for outbound traffic from the WAF to the origin server. These name value pairs are then available to the application.

Securing Your WAF

To secure your WAF, you must configure your servers to accept traffic from the WAF servers. Configure your origin's ingress rules to only accept connections from the following CIDR ranges:

- 192.157.18.0/23
- 205.147.88.0/21
- 192.69.118.0/23
- 198.181.48.0/21
- 199.195.6.0/23

Using the Console

To add an origin to your WAF policy

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to add an origin to. The WAF Policy overview appears.
3. Click **Origin Management**.
4. Click **Add Origin**.

5. In the **Add Origin** dialog box, enter the following:
    - **Origin Name**: A unique name for the origin. Avoid entering confidential information.
    - **URI**: The IPv4 address or fully qualified domain name (FQDN) of the origin. The URI can be a full URI, not just a host/IP.
    - **Set as WAF origin?**: Enable this field to designate this origin as the WAF origin.
    - **HTTPS PORT**: The port used for secure HTTP connection. The default is 443.
    - **HTTP PORT**: The HTTP port the origin listens on. The default is 80.
    - **Header(s)**: (Optional) The name displayed in the HTTP request header and the header value that can be added and passed to the origin server with all requests. Additional headers can be added here.
6. Click **Add Origin**. The origin is added to the Origin Management list. You can now configure WAF rules.

To edit an origin

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to edit the origin for. The WAF Policy overview appears.
3. Click **Origin Management**.
4. In the Origin Management view, select the check box for the origin you want to edit.
5. Click **Edit**.
6. In the Edit Origin dialog box, make the needed changes.
7. Click **Save Origin**.

To delete an origin

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to delete an origin from. The WAF Policy overview appears.
3. Click **Origin Management**.
4. In the Origin Management view, select the check box for the origin you want to delete.
5. Click **Delete**.
6. In the confirmation dialog box, click **Delete**.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

- CreateWaasPolicy
- GetWaasPolicy
- UpdateWaasPolicy To remove an origin from the policy using the API, use the `UpdateWaasPolicy` method and leave the origin field empty upon update.

Each origin has a unique name (key). The name of the origin to be used by the WAF must be referenced in the `wafConfig` portion of the settings. For example, if you have the following origins in your configuration:

```
{

"compartmentId": "ocid1.compartment.oc1..aaaaatsdfssdfsdsdfsgxz",
"lifecycleState": "ACTIVE",
"displayName": "myWAFprotectedApp",
"origins": {
"primaryorigin": {
"httpPort": 80,
"httpsPort": 443,
```

```
                "uri": "67.205.161.231",
                "customHeaders": []
                },

                "secondaryorigin": {
                "httpPort": 80,
                "httpsPort": 443,
                "uri": "54.175.154.7",
                "customHeaders": [
                {
                "name": "OriginHeader",
                "value": "true"
                },
                {
                "name": "OriginHeader2",
                "value": "true"
                }
                ]
        }
```

Then within the `wafConfig`, the origin in use would be referenced by name:

```
                "wafConfig": {
                "deviceFingerprintChallenge": {"isEnabled": false},
                "origin": "secondaryorigin",
                "whitelists": [],
```

In this example, the WAF is actively using `secondaryorigin`.

Bot Management

Bot Management enables you to mitigate undesired bot traffic from your site using CAPTCHA and JavaScript detection tools, while enabling known published bot providers to bypass these controls.

Non-human traffic makes up most of the traffic to sites. Bot Manager is designed to detect and block, or otherwise direct, non-human traffic that may interfere with site operations. The Bot Manager features mitigate bots that conduct content and price scraping, vulnerability scanning, comment spam, brute force attacks, and application-layer DDoS attacks. You can also whitelist good bots.



Warning

When you enable Bot Management, you incur a higher rate on requests to the WAF.

JavaScript Challenge

JavaScript Challenge validates that the client can accept JavaScript with a binary decision. JavaScript Challenge is generally the first level of bot mitigation, but not sufficient with more advanced bot tools, which require more advanced challenges. Additional functionality, like detecting Network Address Translation (NAT) traffic, can mitigate the risk of blocking legitimate user traffic from users behind a shared IP address.

The **Action Threshold** parameter defines the number of requests that fail the challenge before the action is taken. The requests that fail under this threshold are not logged. For example, if you set the JavaScript challenge action to **Block** and the **Action Threshold** to 10, and a client that doesn't accept JavaScript makes 11 requests within the **Action Expire Time**, the first 10 requests will be allowed through to origin (assuming there are no other rules) and logs will show one **Block** entry action taken for the JavaScript Challenge.

CAPTCHA Challenge

If a specific URL should be accessed only by a human, you can control it with CAPTCHA protection. You can customize the comments for the CAPTCHA Challenge for each URL. Bots are kept from accessing protected web application functionality using CAPTCHA images designed to be out of reach of computer vision and OCR technologies.

Good Bot Whitelist

Good Bots provides the list of bots managed by known providers, such as Baidu or Google. You can allow the access from a specific good bot, or block the bot if they serve no business purpose. Allowed good bots from this section are whitelisted.

Whitelisted bots are flagged with a Bypass action in the WAF policy Logs. You can select the **Bypass** check box from the **Action** filter in Logs to search for the traffic allowed from these

rules. Logged good bot events are categorized as a Threat Intelligence Leads log type, however, they are not a threat when the action taken is to Bypass.

The list of good bots on this menu are managed and continuously updated. Additional good bots can be added as a new access control rule in Access Control.

Using the Console

To configure JavaScript Challenge settings

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to configure JavaScript Challenge settings for. The WAF Policy overview appears.
3. Click **Bot Management**.
4. Click **Enable JavaScript Challenge**.
5. In the JavaScript Challenge dialog box, select the **Enable JavaScript Challenge** check box.
6. In the JS Challenge Action section, choose one of the following methods:
    - **Detect Only:** Select this option if you want to be alerted for every matched request.
        - **Set Header for Failed Request:** (Optional) Select this check box to specify an additional HTTP header to requests that fail the challenge.
    - **Block:** Select this option to block requests by returning a response code, error page, or CAPTCHA.
        - **Block Action**: Select the action that will be taken when a matching request is blocked.
        - **Block Response Code:** Select a status code to return in response to blocked requests.
7. Enter the following information:

- **Action Threshold**: Specify the number of failed requests before taking action.
- **Action Expire Time**: Enter the number of seconds between challenges to the same IP address.

8. Click **Save**.

The JavaScript Challenge is added to the list of changes to be published.

To edit JavaScript Challenge settings

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to edit JavaScript Challenge settings for. The WAF Policy overview appears.
3. Click **Bot Management**.
4. Click **Edit JavaScript Challenge**.
5. In the **Edit JavaScript Challenge** dialog box, make the needed changes.
6. Click **Save**.

To add a CAPTCHA Challenge

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to edit CAPTCHA challenge settings for. The WAF Policy overview appears.
3. Click **Bot Management**.
4. Click the **CAPTCHA Challenge** tab.
5. Click **Add CAPTCHA Challenge**.

6. In the Add CAPTCHA Challenge dialog box, enter the following information:
    - **CAPTCHA Title**: Enter the text for the CAPTCHA page title.
    - **CAPTCHA URL Path**: Enter the URL path challenged by CAPTCHA.
    - **Session Duration**: Enter the number of seconds after which the CAPTCHA challenge cannot be resubmitted to the same user.
    - **CAPTCHA Header**: Enter the text that will appear before the CAPTCHA image (for example, "I am not a robot").
    - **Footer Text:** Enter the text that will be shown after the CAPTCHA input box and before the submit button.
    - **Incorrect CAPTCHA Text**: Enter the text that will appear when incorrect text is entered (for example, "The CAPTCHA was incorrect. Please try again.").
    - **Submit button**: Enter the text for the Submit button (for example, "Yes, I am human.").
7. Click **Preview CAPTCHA** to preview the CAPTCHA challenge in a new tab.
8. Click **Add**.

To edit a CAPTCHA Challenge

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to edit CAPTCHA Challenge settings for. The WAF Policy overview appears.
3. Click **Bot Management**.
4. Click the **CAPTCHA Challenge** tab.
5. Select the check box for the CAPTCHA you want to edit.
6. Select **Edit** from the **Actions** drop-down menu.
7. Update the CAPTCHA Challenge and then click **Save**.

To delete a CAPTCHA Challenge

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to delete CAPTCHA Challenge settings for. The WAF Policy overview appears.
3. Click **Bot Management**.
4. Click the **CAPTCHA Challenge** tab.
5. Select the check box for the CAPTCHA Challenge you want to delete.
6. Select **Delete** from the **Actions** drop-down menu.
7. In the Confirm dialog box, click **Delete**.

The deleted CAPTCHA Challenge is added to the list of changes to be published.

To manage the Good Bots Whitelist

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to configure Bot Management for. The WAF Policy overview appears.
3. Click **Bot Management**.
4. Click the **Good Bots Whitelist** tab.
5. Select each bot you want to designate as a good bot.

The designated good bots are added to the list of changes to be published.

To publish changes

Updates to your WAF policy appear in the list to be published in Unpublished Changes. Pending changes do not persist across browser sessions. Once you publish changes, it cannot be edited until changes propagate to the edge nodes.

1. Under WAF Policy, click **Unpublished Changes**.
2. In the Unpublished Changes list, click the drop-down arrow beside an unpublished change to review the change.
3. Click **Publish All**.
4. In the Publish Changes dialog box, click **Publish All**.

To discard changes

Updates to your WAF policy appear in the list to be published in Unpublished Changes.

1. Under WAF Policy, click **Unpublished Changes**.
2. In the Unpublished Changes list, click the drop-down arrow beside an unpublished change to review the change.
3. Select the check box for the change you want to discard.
4. Click **Discard**.
5. In the Discard Change dialog box, click **Discard**.

Using the CLI

You can use the CLI to enable rate limiting, device fingerprinting, and human interaction challenges.

To enable rate limiting

Open a command prompt and run the following command to enable rate limiting:

```
oci waas address-rate-limiting update-waf --is-enabled true  --allowed-rate-per-address 1 --max-delayed-count-per-address 2 --waas-policy-id <policy_ocid>
```

This default rate limit setting will allow one request per second before starting to delay. It will delay for two requests until the traffic falls within the threshold boundaries. It will use the default error response code of 503.

To enable device fingerprinting to detect

Open a command prompt and run the following command to enable device fingerprinting to detect:

```
oci waas device-fingerprint-challenge update --is-enabled true --action DETECT --failure-threshold 2 --
action-expiration-in-seconds 240 --failure-threshold-expiration-in-seconds 600 --max-address-count 2 --
max-address-count-expiration-in-seconds 255 --waas-policy-id <policy_ocid>
```

To enable the human interaction challenge to detect

Open a command prompt and run the following command to enable the human interaction challenge to detect:

```
oci waas human-interaction-challenge update --is-enabled true  --waas-policy-id <policy_ocid>
```

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

- UpdateJsChallenge
- GetJsChallenge
- UpdateCaptchas
- GetCaptchas
- GetDeviceFingerprintChallenge
- UpdateDeviceFingerprintChallenge
- GetHumanInteractionChallenge
- UpdateHumanInteractionChallenge
- GetWafAddressRateLimiting
- UpdateWafAddressRateLimiting

WAF Protection Rules

Protection rules match web traffic to rule conditions and determine the action to be taken when the conditions are met. Protection Rule Settings allow you to define the parameters for enforcement any time a protection rule is matched. Recommendations aid in the optimization of your WAF security profile. The Security Operations team proactively monitors all events to provide recommendations about the action of a specific ruleset. See Supported Protection Rules for additional information.

Using the Console

To apply an action to a protection rule

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to configure rule settings for. The WAF Policy overview appears.
3. Click **Protection Rules**.
4. Click the **Rules** tab.
5. Select the protection rule you want to apply an action to.



Tip

You can use the **Rule ID** or **Rule Action** filters to locate a protection rule.

6. Select one of the following actions from the **Actions** drop-down menu:
   - **Detect**: Matching requests generate an alert and the request is proxied.
   - **Block**: Matching requests are blocked.
   - **Off**: The rule is disabled.

The protection rule action is added to the list to be published.

To edit rule settings

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to configure rule settings for. The WAF Policy overview appears.
3. Click **Protection Rules**.
4. Click the **Rule Settings** tab.
5. Click **Edit Rule Settings**.
6. In the Edit Rule Settings dialog box, enter the following:
    - **Block Action**: The action taken on malicious requests blocked by WAF.
    - **Block Response Code**: Provides information indicating why the request was blocked.
    - **Max Number of Arguments**: The maximum number of arguments allowed in the request. The recommended setting is 255.
    - **Max Length of Argument**: The maximum argument length allowed in the request. The recommended setting is 400.
    - **Max Total Argument Length**: The maximum argument length for all arguments in the request. The recommended setting is 64000.
    - **Recommendations Period**: The period in days to analyze for recommended actions.
    - **Allowed HTTP Methods**: The list of allowed HTTP protocol methods.
7. Click **Save**.

The accepted protection rules are added to the list to be published.

To accept recommendations

Recommendations will begin appearing once sufficient traffic has gone through the WAF to profile the right security posture.

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to configure rule settings for. The WAF Policy overview appears.
3. Click **Protection Rules**.
4. Click the **Recommendations** tab.
5. Select the protection rules you want to accept.



Tip

You can use the **Recommended Action** filter to locate a recommendation by **Detect** or **Block**.

6. Click **Accept Recommendations**.

The accepted protection rules are added to the list to be published.

To publish changes

Updates to your WAF policy appear in the list to be published in Unpublished Changes. Pending changes do not persist across browser sessions. Once you publish changes, it cannot be edited until changes propagate to the edge nodes.

1. Under WAF Policy, click **Unpublished Changes**.
2. In the Unpublished Changes list, click the drop-down arrow beside an unpublished change to review the change.

3. Click **Publish All**.
4. In the Publish Changes dialog box, click **Publish All**.

To discard changes

Updates to your WAF policy appear in the list to be published in Unpublished Changes.

1. Under WAF Policy, click **Unpublished Changes**.
2. In the Unpublished Changes list, click the drop-down arrow beside an unpublished change to review the change.
3. Select the check box for the change you want to discard.
4. Click **Discard**.
5. In the Discard Change dialog box, click **Discard**.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

- GetProtectionRule
- ListProtectionRules
- UpdateProtectionRules
- GetProtectionSettings
- UpdateProtectionSettings
- ListRecommendations
- AcceptRecommendations

Listing and Accepting Protection Rule Recommendations

Use the following operations to get the list of recommended rules:

- ListRecommendations

```
{
   "name": "SQL authentication bypass attempts",
   "action": "OFF",
   "description": "Detects basic SQL authentication bypass attempts.",
   "exclusions": [],
   "key": "981244",
   "tags": "SQL Injections, Recommended"
   },

   {

   "modSecurityRuleIds": [
      "950001",
      "959070",
      "959071",
      "959072",
      "950908",
      "959073"
                              ],

   "name": "Common SQL Injections",
   "action": "OFF",
   "description": "detects common SQL injection attacks",
   "exclusions": [],
   "key": "950001",
   "tags": "SQL Injections, WASCTC, OWASP, A1, PCI, Recommended"

   },
```

Using the key values from the output of the GET call above, you can accept one or more of the recommendations using the following operation passing an array of the keys:

- AcceptRecommendations

Body:

```
[
   “981244”,
```

```
   "950001"
]
```

Protection Rule Specific Settings

Several protection rule settings are settings for specific protection rules.

Setting	Rule ID	Rule Name
Allowed HTTP Methods	911100	Restrict HTTP Request Methods
Max Total Argument Length	960341	Total Arguments Limits
Max Number of Arguments	960335	Number of Arguments Limits
Max Length of Argument	960208	Values Limits

The term "Arguments" refers to either query parameters or body parameters in a PUT/POST request. For instance, if the Max Number of Arguments is 2 and RuleID 960335 is set to BLOCK, any of the following requests would be blocked:

```
GET /myapp/path?query=one&query=two&query=three
```

```
POST /myapp/path with Body {"arg1":"one","arg2":"two","arg3":"three"}
```

```
POST /myapp/path?query=one&query=two with Body {"arg1":"one"}
```

Max Length of Argument is the length of either a name or the value of the argument. **Total Argument Length** refers to the sum of the name and value length.

Exclusions

Sometimes a protection rule can trigger a false positive. You can configure an exception if the request(s) generating the false positive have a particular argument or cookie that can be used to identify that request be excluded from the action normally taken on the rule. Exclusions have to be created through the API. The following exclusion parameters can be used:

Name	Value
REQUEST_COOKIES	Cookie Value
REQUEST_COOKIES_NAMES	Cookie Name (value is irrelevant)
ARGS	Argument (Query Parameter or POST/PUT data)
ARGS_NAMES	Query Parameter Name (value is irrelevant)

EXAMPLE

In this example, a block is applied to WAF Rule 911100 (Restrict HTTP Request Methods) with an exception to allow requests with an argument that contains "passthrough".

```
PUT / waasPolicies /<policy_ocid>/wafConfig/protectionRules
```

With the body:

```
[
 {
       "key":"911100",
       "action":"BLOCK",
       "exclusions":
        [
                {
                 "target":"REQUEST_COOKIES",
                 "exclusions":["yourcompany.com", "Wed, 21 Oct 2015 07:28:00 GMT", "12345", "219ffwef9w0f"]
                },
                {
                                "target":"REQUEST_COOKIES_NAMES",
                                "exclusions":["OAMAuthnCookie", "JSESSIONID", "HCM-PSJSESSIONID"]
                },
                {
                 "target":"ARGS",
                 "exclusions":["passthrough"]
                }
        ]
 }
]
```

This will return a 202 Accepted HTTP status, which means the policy will enter an UPDATING state until changes are provisioned to the edge nodes.

Supported Protection Rules

The Oracle Cloud Infrastructure WAF service supports many protection rule types. The following list provides a brief explanation of the purpose of each protection rule type.

PROTECTION RULES

Rule ID/Key	Name	Description
90001	Filter Profanity	Detects profanity used in request headers and body.
90004	Executable File Upload Attempt	Detects attempts to upload executable files though input forms.
90006	Credit Card Leakage in Request: GSA SmartPay	Detects GSA SmartPay credit card numbers in user input.
90007	Credit Card Leakage in Request: MasterCard	Detects MasterCard credit card numbers in user input.
90008	Credit Card Leakage in Request: Visa	Detects Visa credit card numbers in user input.
90009	Credit Card Leakage in Request: American Express	Detects American Express credit card numbers in user input.
90010	Credit Card Leakage in Request: Diners Club	Detects Diners Club credit card numbers in user input.
90011	Credit Card Leakage in Request: enRoute	Detects enRoute credit card numbers in user input.

Rule ID/Key	Name	Description
90012	Credit Card Leakage in Request: Discover	Detects Discover credit card numbers in user input.
90013	Credit Card Leakage in Request: JCB	Detects JCB credit card numbers in user input.
120123	Joomla! Core CVE-2015-8562 Remote Code Execution Vulnerability Prevention	Detects Joomla! Core CVE-2015-8562 Remote Code Execution Vulnerability payload.
900032	HTTP Parameter Pollution (HPP) Detection	Detects requests that have multiple arguments with the same name indicative of an HPP attack.
911100	Restrict HTTP Request Methods	Allows only request methods specified by the configurable "Allowed http methods" parameter.
920100	Invalid HTTP Request Line	Detects an invalid HTTP request line.
920280	Missing/Empty Host Header	Detects a missing/empty host header.
920350	Invalid HTTP Request Line	Detects invalid HTTP request lines.
941100	Cross-Site Scripting (XSS) Attempt: Libinjection - XSS Detection	Detects XSS Libinjection attempt.
941101	Cross-Site Scripting (XSS) Attempt: SS Attack Detected via libinjection	Detects an SS attack via libinjection.
941110	Cross-Site Scripting (XSS) Attempt: XSS Filters - Category 1	Detects script tag-based XSS vectors, for example, <script> alert(1)</script>.

Rule ID/Key	Name	Description
941120	Cross-Site Scripting (XSS) Attempt: XSS Filters - Category 2	Detects XSS vectors making use of event handlers like onerror, onload etc., for example, <body onload="alert(1)">.
941130	Cross-Site Scripting (XSS) Attempt: XSS Filters - Category 3	Detects XSS vectors making use of attribute vectors.
941140	Cross-Site Scripting (XSS) Attempt: XSS Filters - Category 4	Detects XSS vectors making use of javascript URI and tags, for example, <p style="background:url(javascript:alert(1))">.
941150	Cross-Site Scripting (XSS) Attempt: XSS Filters - Category 5	Detects HTML attributes - src, style, and href.
941160	Cross-Site Scripting (XSS) Attempt: NoScript XSS Filters	Detects NoScript InjectionChecker: HTML Injection.
941170	Cross-Site Scripting (XSS) Attempt: NoScript XSS Filters	Detects NoScript InjectionChecker: Attributes injection.
941180	Cross-Site Scripting (XSS) Attempt: Blacklist Keywords from Node-Validator	Detects Blacklist Keywords from Node-Validator.
941190	Cross-Site Scripting (XSS) Attempt: XSS Filters from Internet Explorer	Detects XSS Filters from IE.

Rule ID/Key	Name	Description
941200	Cross-Site Scripting (XSS) Attempt: XSS Filters from Internet Explorer	Detects XSS Filters from IE.
941210	Cross-Site Scripting (XSS) Attempt: XSS Filters from Internet Explorer	Detects XSS Filters from IE.
941220	Cross-Site Scripting (XSS) Attempt: XSS Filters from Internet Explorer	Detects XSS Filters from IE.
941230	Cross-Site Scripting (XSS) Attempt: XSS Filters from Internet Explorer	Detects XSS Filters from IE.
941240	Cross-Site Scripting (XSS) Attempt: XSS Filters from Internet Explorer	Detects XSS Filters from IE.
941250	Cross-Site Scripting (XSS) Attempt: XSS Filters from Internet Explorer	Detects XSS Filters from IE.
941260	Cross-Site Scripting (XSS) Attempt: XSS Filters from Internet Explorer	Detects XSS Filters from IE.
941270	Cross-Site Scripting (XSS) Attempt: XSS Filters from Internet Explorer	Detects XSS Filters from IE.

Rule ID/Key	Name	Description
941280	Cross-Site Scripting (XSS) Attempt: XSS Filters from Internet Explorer	Detects XSS Filters from IE.
941300	Cross-Site Scripting (XSS) Attempt: XSS Filters from Internet Explorer	Detects XSS Filters from IE.
941310	Cross-Site Scripting (XSS) Attempt: US-ASCII encoding bypass listed on XSS filter evasion	Cross-Site Scripting (XSS) Attempt: US-ASCII encoding bypass listed on XSS filter evasion
941320	Cross-Site Scripting (XSS) Attempt: HTML Tag Handler	Cross-Site Scripting (XSS) Attempt: HTML Tag Handler
941330	Cross-Site Scripting (XSS) Attempt: XSS Filters from Internet Explorer	Detects XSS Filters from IE.
941340	Cross-Site Scripting (XSS) Attempt: XSS Filters from Internet Explorer	Detects XSS Filters from IE.
941350	Cross-Site Scripting (XSS) Attempt: UTF-7 encoding XSS filter evasion for IE	Cross-Site Scripting (XSS) Attempt: UTF-7 encoding XSS filter evasion for IE.
950002	Common System Command Access Attempt	Detects access attempts to common system commands, such as map, telnet, ftp, rcms, cmd.

Rule ID/Key	Name	Description
950005	Common System Files Access Attempt	Detects access attempts to common system files, such as access, passwd, groupm global.asa, httpd.conf, boot.ini, /etc.
950006	Injection for Common System Commands	Detects injections for common system commands such as telnet, map, blocalgroup, ftp, rcmd, echo, cmd, chmod, passwd, mail.
950007	Blind SQL Injection	Detects common blind SQL injection attacks.
950009	Session Fixation	Detects Session Fixation, an attack technique that forces a user's session ID to an explicit value. Depending on the functionality of the target website, several techniques can be utilized to "fix" the session ID value. These techniques range from Cross-site Scripting exploits to peppering the website with previously made HTTP requests. After a user's session ID has been fixed, the attacker will wait for that user to log in. Once the user does so, the attacker uses the predefined session ID value to assume the same online identity.
950010	LDAP Injection	Detects common LDAP data constructions injections.
950011	SSI Injection	Detects common Server-Side-Include format data injections.

Rule ID/Key	Name	Description
950012	HTTP Request Smuggling	Detects specially crafted requests that under certain circumstances could be seen by the attacked entities as two different sets of requests. This allows certain requests to be smuggled through to a second entity without the first one realizing it.
950018	UPDF XSS Injection	Detects submitted links that contains the # fragment in a query_string.
950019	Email Injection	Detects mail command injections targeting mail servers and webmail applications that construct IMAP/SMTP statements from user-supplied input that is not properly sanitized.
950103	Path/Directory Traversal	Detects path traversal attempts, also known as directory traversal or "../" attacks.
950107	URL Encodings Validation	Detects URL encoding inconsistencies, encoding abuse, and invalid formatting.
950116	Unicode Encoding/Decoding Validation	blocks full-width Unicode encoding as decoding evasions could be possible.
950117	URL Contains an IP Address	Detects a common RFI attack, when a URL contains an IP address.
950118	PHP Include() Function	Detects a common RFI php include() function attacks.
950119	Data Ends with Question Mark (s) (?)	Detects a common RFI attack, when data ends with question mark(s) (?).

Rule ID/Key	Name	Description
950120	Host Doesn't Match Localhost	Detects a common RFI attack, when host doesn't match localhost.
950801	UTF Encoding Validation	Detects UTF encoding inconsistencies and invalid formatting.
950907	OS Command Injection	Detects OS command injection in an application to elevate privileges, execute arbitrary commands, compromise the underlying operating system and install malicious toolkits such as those to participate in botnet attacks.
950910	HTTP Response Splitting	Detects Carriage Return + Linefeed characters in the response header that could cause attacked entities to interpret it as two separate responses instead of one.
958000	Addimport XSS Attack	Detects usage of addimport in request, cookies, or arguments.
958001	document Cookie XSS Attack	Detects usage of document.cookie in request, cookies, or arguments.
958002	execscript XSS Attack	Detects usage of execscript in request, cookies, or arguments.
958003	fromcharcode XSS Attack	Detects usage of fromcharcode in request, cookies, or arguments.
958004	innerhtml XSS Attack	Detects usage of innerhtml in request, cookies, or arguments.

Rule ID/Key	Name	Description
958005	cdata XSS Attack	Detects usage of cdata in request, cookies, or arguments.
958006	body background XSS Attack	Detects usage of <body background in request, cookies, or arguments.
958007	onload XSS Attack	Detects usage of onload in request, cookies, or arguments.
958008	input type image XSS Attack	Detects usage of <input type image in request, cookies, or arguments.
958009	import XSS Attack	Detects usage of import in request, cookies, or arguments.
958010	activexobject XSS Attack	Detects usage of activexobject in request, cookies, or arguments.
958011	background-image: XSS Attack	Detects usage of background-image: in request, cookies, or arguments.
958012	copyparentfolder XSS Attack	Detects usage of copyparentfolder in request, cookies, or arguments.
958013	createtextrange XSS Attack	Detects usage of createtextrange in request, cookies, or arguments.
958016	getparentfolder XSS Attack	Detects usage of getparentfolder in request, cookies, or arguments.
958017	getspecialfolder XSS Attack	Detects usage of getspecialfolder in request, cookies, or arguments.

Rule ID/Key	Name	Description
958018	href javascript: XSS Attack	Detects usage of href javascript: in request, cookies, or arguments.
958019	href schell XSS Attack	Detects usage of href schell in request, cookies, or arguments.
958020	href vbscript: XSS Attack	Detects usage of href vbscript: in request, cookies, or arguments.
958022	livescript: XSS Attack	Detects usage of livescript: in request, cookies, or arguments.
958023	lowsrc javascript: XSS Attack	Detects usage of lowsrc javascript: in request, cookies, or arguments.
958024	lowsrc shell XSS Attack	Detects usage of lowsrc shell in request, cookies, or arguments.
958025	lowsrc vbscript XSS Attack	Detects usage of lowsrc vbscript in request, cookies, or arguments.
958026	mocha: XSS Attack	Detects usage of mocha: in request, cookies, or arguments.
958027	onabort XSS Attack	Detects usage of onabort in request, cookies, or arguments.
958028	settimeout XSS Attack	Detects usage of settimeout in request, cookies, or arguments.
958030	src http: XSS Attack	Detects usage of src http: in request, cookies, or arguments.

Rule ID/Key	Name	Description
958031	javascript: XSS Attack	Detects usage of javascript: in request, cookies, or arguments.
958032	src and shell XSS Attack	Detects usage of src and shell in request, cookies, or arguments.
958033	vbscript: XSS Attack	Detects usage of vbscript: in request, cookies, or arguments.
958034	style bexpression XSS Attack	Detects usage of style bexpression in request, cookies, or arguments.
958036	type application x-javascript XSS Attack	Detects usage of type application x-javascript in request, cookies, or arguments.
958037	type application x-vbscript XSS Attack	Detects usage of type application x-vbscript in request, cookies, or arguments.
958038	type text ecmascript XSS Attack	Detects usage of type text ecmascript in request, cookies, or arguments.
958039	type text javascript XSS Attack	Detects usage of type text javascript in request, cookies, or arguments.
958040	type text jscript XSS Attack	Detects usage of type text jscript in request, cookies, or arguments.
958041	type text vbscript XSS Attack	Detects usage of type text vbscript in request, cookies, or arguments.
958045	url javascript: XSS Attack	Detects usage of url javascript: in request, cookies, or arguments.

Rule ID/Key	Name	Description
958046	url shell XSS Attack	Detects usage of <url shell in request, cookies, or arguments.
958047	url vbscript: XSS Attack	Detects usage of url vbscript: in request, cookies, or arguments.
958049	?meta XSS Attack	Detects usage of ?meta in request, cookies, or arguments.
958051	?script XSS Attack	Detects usage of < ?script in request, cookies, or arguments.
958052	alert XSS Attack	Detects usage of alert in request, cookies, or arguments.
958054	lowsrc and http: XSS Attack	Detects usage of lowsrc and http: in request, cookies, or arguments.
958056	iframe src XSS Attack	Detects usage of iframe src in request, cookies, or arguments.
958057	?iframe XSS Attack	Detects usage of ?iframe in request, cookies, or arguments.
958059	asfunction: XSS Attack	Detects usage of asfunction: in request, cookies, or arguments.
958291	Range Header Validation	Detects range header inconsistencies and invalid formatting.
958295	Connection Header Validation	Detects connection header inconsistencies and invalid formatting.

Rule ID/Key	Name	Description
958404	onerror XSS Attack	Detects usage of onerror in request, cookies, or arguments.
958405	onblur XSS Attack	Detects usage of onblur in request, cookies, or arguments.
958406	onchange XSS Attack	Detects usage of onchange in request, cookies, or arguments.
958407	onclick XSS Attack	Detects usage of onclick in request, cookies, or arguments.
958408	ondragdrop XSS Attack	Detects usage of ondragdrop in request, cookies, or arguments.
958409	onfocus XSS Attack	Detects usage of onfocus in request, cookies, or arguments.
958410	onkeydown XSS Attack	Detects usage of onkeydown in request, cookies, or arguments.
958411	onkeypress XSS Attack	Detects usage of onkeypress in request, cookies, or arguments.
958412	onkeyup XSS Attack	Detects usage of onkeyup in request, cookies, or arguments.
958413	onload XSS Attack	Detects usage of onload in request, cookies, or arguments.
958414	onmousedown XSS Attack	Detects usage of onmousedown in request, cookies, or arguments.

Rule ID/Key	Name	Description
958415	onmousemove XSS Attack	Detects usage of onmousemove in request, cookies, or arguments.
958416	bonmouseout XSS Attack	Detects usage of bonmouseout in request, cookies, or arguments.
958417	bonmouseover XSS Attack	Detects usage of bonmouseover in request, cookies, or arguments.
958418	onmouseup XSS Attack	Detects usage of onmouseup in request, cookies, or arguments.
958419	onmove XSS Attack	Detects usage of onmove in request, cookies, or arguments.
958420	onresize XSS Attack	Detects usage of onresize in request, cookies, or arguments.
958421	onselect XSS Attack	Detects usage of onselect in request, cookies, or arguments.
958422	onsubmit XSS Attack	Detects usage of onsubmit in request, cookies, or arguments.
958423	onunload XSS Attack	Detects usage of onunload in request, cookies, or arguments.
959151	php Code Injection	Detects a common injections attack, when request contain any php code e.g. "<\?>".
960000	File Name Validation	Detects multipart/form-data file name evasion attempts.

Rule ID/Key	Name	Description
960007	Missing Host Header	Detects missing request host header.
960009	Missing User-Agent Header	Detects missing request user-agent header.
960011	GET/HEAD Requests Validation	Detects if GET/HEAD requests contain request body since it is not a common practice.
960012	Content-Length Header Validation	Detects if content-length header is provided with every POST request.
960013	Require Content-Length to be provided with every HTTP/1.1 POST request that has no Transfer-Encoding header	Detects HTTP/1.1 requests that do not comply with HTTP 1.1 spec by having no content-length header when transfer-encoding is also absent.
960014	URI Validation	Ensures that URI and canonical server name are matching.
960015	Missing Accept Header	Detects missing request accept header.
960016	Content-Length Header Validation	Detects if content-length HTTP header is not numeric.
960017	Host Header Is IP Address	Detects if host header is a numeric IP address as it could be indicative of automated client access.
960020	Pragma Header Validation	Ensures that pragma, cache-control headers and HTTP protocol version supplied by the client are matching.
960022	Expect Header Validation	Ensures that expect header and HTTP protocol version supplied by the client are matching.

Rule ID/Key	Name	Description
960024	Repetitive Non-Word Chars	Attempts to identify when 4 or more non-word characters are repeated in sequence.
960208	Values Limits	Detects HTTP requests with value length exceeding the configurable "Max length of argument" parameter.
960209	Arguments Limits	Detects HTTP requests with argument name length exceeding the 100 symbols.
960335	Number of Arguments Limits	Detects HTTP requests with number of arguments exceeding the configurable "Max amount of arguments" value.
960341	Total Arguments Limits	Detects HTTP requests with total length of all arguments exceeding the configurable "Max total argument length" parameter.
960901	Character Set Validation	Ensures that only a specific character set(s) is used.
960902	Content-Encoding Header Validation	Ensures that identity is not specified in content-encoding header.
960904	Missing Content-Type Header	Detects missing content-type header or if combination of content-length and content-type headers is invalid.

Rule ID/Key	Name	Description
960911	Request Line Format Validation Against the HTTP RFC	Uses rule negation against the regex for positive security. The regex specifies the proper construction of URI request lines such as: "http:" "//" host [":" port] [abs_path ["?" query]]. It also outlines proper construction for CONNECT, OPTIONS, and GET requests.
960912	Malformed request bodies	Checks for request body parsing errors.
960914	Strict Multipart Parsing Checks	Uses strict checks for what is accepted in the multipart/form-data request body. If the rule proves to be too strict for your environment consider changing it to Off.
960915	Multipart Unmatched Boundary Check	Checks for signs of evasions during file upload requests.
970901	5XX Status Code Information Leakage	Detects if an application generates 500-level status code. For example, 500 Internal Server Error, 501 Not Implemented...505 HTTP Version Not Supported.
973300	Common Direct HTML Injection	Detects tags that are the most common direct HTML injection points.
973306	Embedded JavaScript in Style Attribute	Detects embedded JavaScript in style attribute.
973307	Embedded Scripts Within JavaScript Fragments	Detects common JavaScript fragments like fromcharcode, alert, eval that can be used for attacks.

Rule ID/Key	Name	Description
973309	CSS Fragments Attacks	Detects common CSS fragments attacks like or <img style="x:expression (document.write(1))">.
973310	Embedded Scripts Within Alert Fragments	Detects attacks like alert('xss'), alert("xss"), alert(/xss/).
973311	String.fromCharCode (88,83,83) attacks	Detects String.fromCharCode(88,83,83) attacks.
973312	'';!--"<XSS>=&{()} Attacks	Detects '';!--"<XSS>=&{()} attacks.
973313	&{alert('xss')} Attacks	Detects &{alert('xss')} attacks.
973314	Doctype Entity Inject	Detects Doctype Entity inject attacks.
973331	Internet Explorer XSS Filters	Detects common IE XSS attacks.
973336	Embedding Scripts Within Scripts	Detects script tag-based XSS vectors. For example, <script> alert(1)</script>.
973337	Embedded Scripts Within Event Handlers	Detects event handler based XSS vectors. For example, <body onload="alert(1)">.
973338	Embedded Scripts Within URI Schemes	Detects "data", "javascript", "src" or other URI schemes/attributes based XSS vectors. For example, <p style="background:url (javascript:alert(1))">.

Rule ID/Key	Name	Description
981136	Generic XSS Attacks	Detects common XSS attacks embedded within non-script elements. For example, jscript onsubmit copyparentfolder document javascript meta onchange onmove onkeydown onkeyup activexobject onerror onmouseup ecmascript bexpression onmouseover vbscript.
981172	SQL Character Anomaly Scoring	Attempts to gauge when there is an excessive use of meta-characters within a single parameter payload.
981227	Request URI Validation	Detects invalid URI in request.
981242	–°lassic SQL Injection Probings	Detects classic SQL injection probings.
981244	SQL Authentication Bypass Attempts	Detects basic SQL authentication bypass attempts.
981245	SQL Authentication Bypass Attempts	Detects basic SQL authentication bypass attempts.
981246	SQL Authentication Bypass Attempts	Detects basic SQL authentication bypass attempts.
981272	SQL Injection Using sleep() or benchmark()	Detects blind SQL injection tests using sleep() or benchmark() functions.
981300	SQL Keyword Anomaly Scoring	Detects common SQL keywords anomalies.

Rule ID/Key	Name	Description
981318	String Termination/Statement Ending	Identifies common initial SQLi probing requests where attackers insert/append quote characters to the existing normal payload to see how the app/db responds.
1000000	Shellshock Exploit Attempt	Detects the ability to unintentionally execute commands in Bash (CVE-2014-6271).
2017100	Apache Struts 2 Multipart Parser CVE-2017-5638 Remote Code Execution Vulnerability Prevention	Detects Apache Jakarta CVE-2017-5638 Remote Code Execution Vulnerability Payload.
2018100	CVE-2018-6389 WordPress Parameter Resource Consumption Remote DoS	Detects WordPress parameter resource consumption remote DoS on jquery-ui-core .
2100019	/_layouts/scriptresx.ashx sections Parameter XSS	Detects Microsoft SharePoint /_ layouts/scriptresx.ashx sections parameter XSS attacks.
2100023	/owssrv.dll List Parameter XSS	Detects Microsoft SharePoint /owssrv.dll List Parameter XSS attacks.
2100026	_ layouts/Chart/WebUI/WizardList.aspx skey Parameter XSS	Detects Microsoft SharePoint _ layouts/Chart/WebUI/WizardList.aspx skey Parameter XSS attacks.
2100027	_layouts/themeweb.aspx XSS	Detects Microsoft SharePoint _ layouts/themeweb.aspx ctl00$PlaceHolderMain$ctl82 $customizeThemeSection$accent6 Parameter XSS attacks.

Rule ID/Key	Name	Description
2100028	_layouts/inplview.aspx ListViewPageUrl Parameter XSS	Detects Microsoft SharePoint _ layouts/inplview.aspx ListViewPageUrl Parameter XSS attacks.
2100032	owssrv.dll View Parameter XSS	Detects Microsoft SharePoint owssrv.dll View Parameter XSS attacks.
2100033	NewForm.aspx TextField_ spSave Parameter XSS	Detects Microsoft SharePoint NewForm.aspx TextField_spSave Parameter XSS attacks.
2100034	/Lists/Calendar/calendar.aspx CalendarDate Parameter XSS	Detects Microsoft SharePoint /Lists/Calendar/calendar.aspx CalendarDate Parameter XSS attacks.
2100035	_layouts/Picker.aspx XSS	Detects Microsoft SharePoint _ layouts/Picker.aspx ctl00$PlaceHolderDialogBodySection$ctl04$hiddenSpanData Parameter XSS attacks.
2100048	_layouts/help.aspx cid0 Parameter XSS	Detects Microsoft SharePoint _layouts/help.aspx cid0 Parameter XSS attacks.
2100062	_layouts/ScriptResx.ashx name Parameter LFI	Detects Microsoft SharePoint _ layouts/ScriptResx.ashx name Parameter LFI attacks.
2100063	_ layouts/OSSSearchResults.aspx k Parameter XSS	Detects Microsoft SharePoint _ layouts/OSSSearchResults.aspx k Parameter XSS attacks.

Rule ID/Key	Name	Description
2100069	wiki pages multiple Parameter XSS	Detects Microsoft SharePoint wiki pages multiple Parameter XSS (CVE-2013-3180) attacks.
2100070	/Lists/Links/AllItems.aspx XSS	Detects Microsoft SharePoint /Lists/Links/AllItems.aspx ctl00mg_2085a732_4692_4d3e_99d2_4d90ea5108d2$ctl00$ctl05$ctl00$ctl00$ctl00$ctl04$ctl00$ctl00$UrlFieldUrl Parameter XSS attacks.
2100082	Drupal - pre-auth SQL Injection Vulnerability	Detects Drupal pre-auth SQL injection vulnerability. A malicious user can inject arbitrary SQL queries and thereby control the complete Drupal site. This leads to a code execution as well. Drupal 7.32 fixed this bug.
2100083	Gerber WebPDM XSS Vulnerability	Detects cross-site scripting vulnerability in Gerber WebPDM Product Data Management System.
2100084	Gerber WebPDM SQL Injection Vulnerability	Detects SQL Injection Vulnerability in Gerber WebPDM Product Data Management System.
2100085	High X-SharePointHealthScore	Detects Microsoft SharePoint High X-SharePointHealthScore - potential DoS attack/availability risk.
2100086	Response Header Found	Detects Microsoft SharePoint SharePointError Response Header Found.
2100087	x-virus-infected Response Header Found	Detects x-virus-infected Response Header Found.

Rule ID/Key	Name	Description
2100088	Rights Management (IRM) Error Response Header Found	Detects Microsoft SharePoint Information Rights Management (IRM) Error Response Header Found.
2100089	/_ layouts/mobile/editform.aspx XSS	Detects Microsoft SharePoint /_ layouts/mobile/editform.aspx XSS attacks.
2200924	IRC Botnet Attacks	Detects common IRC Botnet attack commands.
2200925	Detects HOIC DoS Tool Requests	Detects HOIC DoS tool requests.
2250117	Common RFI Attacks	Detects common types of Remote File Inclusion (RFI) attacks.
2250120	Local File Inclusion Attacks	Detects common local file inclusion attacks like my $dir = "../../../../../../../../../../../../"; or "http://".$site.$bug.$dir."/proc/self/environ%0000";.
2250121	Local File Inclusion ENV Attack in User-Agent	Detects Local File Inclusion ENV Attack in User-Agent
2250122	PHP Injection Attack	Detects common php injection attacks like "send-contactus=1&author_name=[php]eval (base64_decode ('".$code."'))%3Bdie%28%29%3B%5B%2Fphp%5D"

Rule ID/Key	Name	Description
2250123	XML-RPC PHP Injection Attack	Detects common XML-RPC PHP Injections like $exploit .= "echo'j13mb0t';".$code."echo'j13mb0t';exit;/* </name> </value></param></params></methodCall>";
2250125	osCommerce File Upload	Detects osCommerce file upload attacks like "http://".$site."admin/file_ manager.php/login.php";.
2250126	Oscommerce File Disclosure and Admin ByPass	Detects Oscommerce File Disclosure and Admin ByPass.
2250127	e107 Plugin my_gallery Exploit	Detects e107 Plugin my_gallery Exploit "http://".$site."e107_plugins/my_ gallery/image.php?file=../../e107_config.php".
2250128	Opencart Remote File Upload Vulnerability	Detects Opencart remote file upload vulnerability.
2250129	Zen Cart Local File Disclosure Vulnerability	Detects Zen Cart local file disclosure vulnerability.
20182056	CVE-2003-1567 CVE-2004-2320 CVE-2010-0360 TRACE & CONNECT Attempts	Detects TRACE method attempts.
20182137 5	CVE-2012-0209, Remote Execution Backdoor Attempt Against Horde	Detects remote execution backdoor attempt against Horde.

Rule ID/Key	Name	Description
201821438	CVE-2012-1723, CVE-2012-1889, CVE-2012-4681, Blackhole Exploit Kit JavaScript Carat String Splitting with Hostile Applet	Detects Blackhole exploit kit JavaScript carat string splitting with hostile applet.
201822063	CVE-2012-1823, CVE-2012-2311, CVE-2012-2335, CVE-2012-2336, PHP-CGI Remote File Include Attempt	Detects PHP-CGI remote file include attempts.
201826834	CVE-2012-4681, CVE-2012-5076, CVE-2013-2423, Sweet Orange Exploit Kit Landing Page in.php base64 uri	Detects Sweet Orange exploit kit landing page in.php base64 uri attacks.
201826947	CVE-2013-2423, DotkaChef/Rmayana/DotCache Exploit Kit Inbound Java Exploit Download	Detects DotkaChef/Rmayana/DotCache exploit kit inbound java exploit download attacks.
201826948	CVE-2013-1493, DotkaChef/Rmayana/DotCache Exploit Kit Inbound Java Exploit Download	Detects DotkaChef/Rmayana/DotCache exploit kit inbound java exploit download attacks.
201827040	CVE-2013-0422, CVE-2013-2423, Styx Exploit Kit Plugin Detection Connection Jorg	Detects Styx exploit kit plugin detection connection jorg attacks.

Rule ID/Key	Name	Description
201841409	CVE-2017-3823, CVE-2017-6753, Cisco WebEx Explicit Use of Web Plugin	Detects Cisco WebEx explicit use of web plug-in.
201843811	CVE-2017-9812, Kaspersky Linux File Server WMC Directory Traversal Attempt	Detects Kaspersky Linux file server WMC directory traversal attempts.

Access Control

As a WAF administrator you can define explicit actions for requests that meet various conditions. Conditions use various operations and regular expressions. A rule action can be set to log and allow, detect, or block requests.

The available conditions are shown in the following table:

Criteria Type	Criteria
URL	Users shall be able define to one or more criteria based on: • URL is • URL is not • URL starts with • URL ends with • URL contains • URL regex The URL regex matching uses Perl-compatible regular expressions.
IP Address	Users shall be able define to one or more criteria based on: • Client IP Address is • Client IP Address is not These values can be a valid IPv4 address, subset, or CIDR notation for a range. IPv6 is not yet supported.
Geo/Country	Users shall be able define to one or more criteria based on: • Country is • Country is not For the API, use a 2-letter country code.

Criteria Type	Criteria
UserAgent	User Agent is a value that identifies the browser client. • User Agent is • User Agent is not
HTTP Header	HTTP Request headers can be evaluated as criteria: • HTTP Header contains The HTTP Header contains value should be entered with colon-delimited <name>:<value> .

You can use the IP Whitelist tab to manage whitelists containing trusted IP addresses that bypass all rules and challenges.

Using the Console



Note

The WAF uses a first-match algorithm so that once an Access Rule criteria matches, it will stop evaluating future rules. The order of rules matters. Use the API to reorder rules.

To add an access rule

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to view IP Address Whitelists for. The WAF Policy overview appears.

3. Click **Access Control**.
4. Select the **Access Rules** tab.
5. Click **Add Access Rule**.
6. In the Add Access Rule dialog box, enter the following:
    - **Name:** A unique name for the access rule. Avoid entering confidential information.
    - **Rule Condition:** Select the condition that must be met before the rule is matched and specify the details of the condition. Additional conditions can be added in this section.
    - **Rule Action:** Determines the response to a request when the rule is matched. Select one of the following options:
        - **Log and Allow**: A log will be created for all matched requests and no further action will be taken.
        - **Detect Only**: A detection alert will be created for all matched requests and no further action will be taken.
        - **Block:** All matched requests will be blocked and a browser page for the selected response code will be returned.
            - **Block Action**: Select the action that will be taken when a matching request is blocked.
            - **Block Response Code:** Select a response code that will be returned when the request has been blocked. The response code provides information indicating why the request was blocked. The default response code is 403 "Forbidden".
7. Click **Add Access Rule**. The access rule is added to the access rule list.

To edit an access rule

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge**

Services and click **WAF Policies**.

2. Click the name of the WAF Policy you want to view access rules for. The WAF Policy overview appears.
3. Click **Access Control**.
4. Select the **Access Rules** tab.
5. Select the check box for the access rule you want to update and then select **Edit** from the **Actions** drop-down menu.
6. In the Edit Access Rule dialog box, make the necessary updates and then click **Save**.

To delete an access rule

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to view access rules for. The WAF Policy overview appears.
3. Click **Access Control**.
4. Select the **Access Rules** tab.
5. Select the check box for the access rule you want to delete and then select **Delete** from the **Actions** drop-down menu.

To add an IP address whitelist

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to view IP Address Whitelists for. The WAF Policy overview appears.
3. Click **Access Control**.
4. Select the **IP Whitelist** tab.

5. Click **Add IP Address Whitelist**.
6. In the Add IP Address Whitelist dialog box, enter the following:
   - **Whitelist Name:** A name for the IP addresses used in the list.
   - **IP Addresses:** Select an IP address or enter an IP address and select it to add it. This field supports CIDR notation.
7. Click **Add IP Address Whitelist**.

   The IP Address Whitelist is added to the list of changes to be published.

To edit an IP Address Whitelist

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to view IP Address Whitelists for. The WAF Policy overview appears.
3. Click **Access Control**.
4. Select the **IP Whitelist** tab.
5. Select the check box for the IP Address Whitelist name you want to edit.
6. Click **Edit**.
7. In the Edit IP Address Whitelist dialog box, make the needed changes.
8. Click **Save**.

   The IP Address Whitelist change is added to the list of changes to be published.

To delete an IP Address Whitelist

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to view alerts for. The WAF Policy overview appears.

3. Click **Access Control**.
4. Select the **IP Whitelist** tab.
5. Select the check box for the IP Address Whitelist name you want to delete.
6. Click **Delete**.

   The deleted IP Address Whitelist is added to the list of changes to be published.

To publish changes

Updates to your WAF policy appear in the list to be published in Unpublished Changes. Pending changes do not persist across browser sessions. Once you publish changes, it cannot be edited until changes propagate to the edge nodes.

1. Under WAF Policy, click **Unpublished Changes**.
2. In the Unpublished Changes list, click the drop-down arrow beside an unpublished change to review the change.
3. Click **Publish All**.
4. In the Publish Changes dialog box, click **Publish All**.

To discard changes

Updates to your WAF policy appear in the list to be published in Unpublished Changes.

1. Under WAF Policy, click **Unpublished Changes**.
2. In the Unpublished Changes list, click the drop-down arrow beside an unpublished change to review the change.
3. Select the check box for the change you want to discard.
4. Click **Discard**.
5. In the Discard Change dialog box, click **Discard**.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operations to get an array of all access rules in the policy:

- UpdateAccessRules
- ListAccessRules
- UpdateWhitelists
- ListWhitelists

To create an access rule:

```
PUT /waasPolicies/{waasPolicyId}/wafConfig/accessRules
```

```
[
   {
   "name": "DetectRequestsToHealthCheck",
   "criteria": [
      {
      "condition": "URL_IS",
      "value": "/health/check"
      }
               ],
   "action": "DETECT",
      }
]
```

Threat Intelligence

WAF has several sources of known IP address threats that are updated daily. The IP address threats are displayed in the following table:

Source	Description
ABUSE|ch	Blacklist of "bad" SSL certificates identified by abuse.ch to be associated with malware or botnet activities.
Bambenek Consulting	Active and non-sinkholed Command & Control (C&C) IP addresses.
BlockList.de	Includes IP addresses for hosting phishing sites and other kinds of fraud activities such as ad-click or gaming fraud.
BruteForceBlocker Project	Feed of known IP addresses from blocked SSH brute force attacks.
Proofpoint ET Labs	IP addresses involved in suspicious and malicious activity.
Feodo IP Blocklist	IP addresses used as C&C communication channel by the Feodo Trojan.
Palevo	IP addresses which are being used as botnet C&C for the Palevo crimeware.
Webroot BotNets	Botnet C&C channels and infected zombie machine controlled by Bot master.
Webroot Denial of Service	Includes DOS, DDOS, anomalous sync flood, and anomalous traffic detection.
Webroot Mobile Threats	IP addresses of malicious and unwanted mobile applications. This category leverages data from the Webroot mobile threat research tea.
Webroot Phishing	IP addresses hosting phishing sites and other kinds of illicit activities such as ad-click or gaming fraud.
Webroot Proxy	IP addresses providing proxy and def services.

Source	Description
Webroot Reputation	IP addresses currently known to be infected with malware. This category also includes IP addresses with an average low Webroot Reputation Index score.
Webroot Scanners	Includes all reconnaissance such as probes, host scan, domain scan and password brute force attacks.
Webroot Spam Sources	Includes tunneling spam messages through proxy, anomalous SMTP activities, and forum spam activities.
Webroot Tor Proxy	Includes IP addresses acting as exit nodes for the Tor Network. Exit nodes are the last point along the proxy chain and make a direct connection to the originator's intended destination.
Webroot Web Attacks	Includes known IP addresses involved in cross site scripting, iFrame injection, SQL injection, cross domain injection, or domain password brute force attacks.
Webroot Windows Exploits	Includes active IP addresses offering or distributing malware, shell code, rootkits, worms or viruses.
ZueS	IP blocklist including known C&C servers/hosts.

Using the CLI

You can use the CLI to enable threat intelligence sources to block.

Open a command prompt and run the following command to list the keys for all of the threat intelligence:

```
oci waas threat-feed list --waas-policy-id <policy_ocid>
```

Then parse the keys to block and add them to the JSON:

```
oci waas threat-feed update --threat-feeds '[{"key":"<key_id>","action":"BLOCK"}]' --waas-policy-id
<policy_ocid>
```

For example:

```
oci waas threat-feed update --threat-feeds '[{"key":"0998d237-bce8-4612-82c8-
a1ca126c0492","action":"BLOCK"}]' --waas-policy-id
ocid1.waaspolicy.oc1..aaaaaaaapfa5zrwnns75kru7mrlzkkmcdevp7w55ld3phjxtgl4s2phuepjq
```

Using the API

Enabling Threat Intelligence can only be performed by using the API at this time.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

To return a set of keys for the threat intelligence:

- ListThreatFeeds



Note

Do not use the keys in the example below, as keys are unique across each policy.

```
{
        "8d3f7f1b-673f-4e3a-ba49-08226f385df3": "OFF",
        "0ff7b308-6afe-4b83-91e0-e3ca04afed6e": "OFF",
        "ea5d7c67-1326-43c9-ac31-1df034b9c063": "OFF",
        "87b420ca-5fbb-4ad4-aeba-1b02a9e60b30": "OFF",
        "2168fc70-2d05-466a-9db5-c13c0e32177d": "OFF",
        "7d080a4a-58ce-4370-a02c-f600b3a84e7b": "OFF",
        "a36c7c50-e99e-4b84-9140-5653fc68ce8d": "OFF",
        "5de7bbc1-313f-4995-9810-f6f77cfd30c9": "OFF",
        "fd2152cc-14f5-4471-a58b-d94cc8a61444": "OFF",
        "cfacd3d3-65d9-4368-93e0-62c906e7a748": "OFF",
        "6eb86368-01ea-4e94-ac1b-49bf0e551443": "OFF",
        "aabb45d9-0d75-481d-9568-58ecad217e1e": "OFF",
        "3805ecc2-1d6d-428b-a03e-2a0fe77fd46f": "OFF",
        "c3452861-4910-4f3a-9872-22cf92d424eb": "OFF",
        "4cf31deb-11af-460e-a46a-ecc1946a6688": "OFF",
        "eff34d63-6235-4081-976d-acd39248bdc3": "OFF",
```

```
      "1d1c94d9-038b-45eb-acd4-fb422e281f4c": "OFF",
      "687b5ff4-b1b6-4d12-8dba-3ea90b4536a1": "OFF",
      "65cf274d-991b-41f8-adda-6fe60ba2704f": "OFF"
}
```

To set all threats to DETECT:

- UpdateThreatFeeds

  With body:

```
[
{"action":"DETECT","key":"8d3f7f1b-673f-4e3a-ba49-08226f385df3"},
{"action":"DETECT","key":"0ff7b308-6afe-4b83-91e0-e3ca04afed6e"},
{"action":"DETECT","key":"ea5d7c67-1326-43c9-ac31-1df034b9c063"},
{"action":"DETECT","key":"87b420ca-5fbb-4ad4-aeba-1b02a9e60b30"},
{"action":"DETECT","key":"2168fc70-2d05-466a-9db5-c13c0e32177d"},
{"action":"DETECT","key":"7d080a4a-58ce-4370-a02c-f600b3a84e7b"},
{"action":"DETECT","key":"a36c7c50-e99e-4b84-9140-5653fc68ce8d"},
{"action":"DETECT","key":"5de7bbc1-313f-4995-9810-f6f77cfd30c9"},
{"action":"DETECT","key":"fd2152cc-14f5-4471-a58b-d94cc8a61444"},
{"action":"DETECT","key":"cfacd3d3-65d9-4368-93e0-62c906e7a748"},
{"action":"DETECT","key":"6eb86368-01ea-4e94-ac1b-49bf0e551443"},
{"action":"DETECT","key":"aabb45d9-0d75-481d-9568-58ecad217e1e"},
{"action":"DETECT","key":"3805ecc2-1d6d-428b-a03e-2a0fe77fd46f"},
{"action":"DETECT","key":"d9cfc537-dd50-427d-830e-a612f535c11f"},
{"action":"DETECT","key":"c3452861-4910-4f3a-9872-22cf92d424eb"},
{"action":"DETECT","key":"4cf31deb-11af-460e-a46a-ecc1946a6688"},
{"action":"DETECT","key":"eff34d63-6235-4081-976d-acd39248bdc3"},
{"action":"DETECT","key":"1d1c94d9-038b-45eb-acd4-fb422e281f4c"},
{"action":"DETECT","key":"687b5ff4-b1b6-4d12-8dba-3ea90b4536a1"},
{"action":"DETECT","key":"65cf274d-991b-41f8-adda-6fe60ba2704f"}
]
```

  This will return a 202 Accepted HTTP status, which means the policy will enter an UPDATING state until changes are provisioned to the edge nodes.

Settings

To use SSL with your WAF policy, you must add a certificate bundle. The certificate bundle you upload includes the public certificate and the corresponding private key. Self-signed

certificates can be used for the internal communication within Oracle Cloud Infrastructure.

Working with SSL Certificates

Oracle Cloud Infrastructure accepts third-party and self-signed certificates in PEM format only. The following is an example PEM encoded certificate:

```
-----BEGIN CERTIFICATE-----
<Base64_encoded_certificate>
-----END CERTIFICATE-----
```

Obtaining Third-Party SSL Certificates

You can purchase an SSL certificate from a trusted Certificate Authority such as Symantec, Thawte, RapidSSL, or GeoTrust. The certificate issuer will provide an SSL certificate that includes a certificate, intermediate certificate, and private key. Use this information, including the intermediate certificate, when adding an SSL certificate to Oracle Cloud Infrastructure.

Converting to PEM format

If you receive your certificates and keys in formats other than PEM, you must convert them before you can upload them to the system. You can use OpenSSL to convert certificates and keys to PEM format.

Uploading Certificate Chains

If you have multiple certificates that form a single certification chain, you must include all relevant certificates in one file before you upload them to the system. The following example of a certificate chain file includes four certificates:

```
-----BEGIN CERTIFICATE-----
<Base64_encoded_certificate>
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
<Base64_encoded_certificate>
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
<Base64_encoded_certificate>
-----END CERTIFICATE-----
```

```
-----BEGIN CERTIFICATE-----
<Base64_encoded_certificate>
-----END CERTIFICATE-----
```

SUBMITTING PRIVATE KEYS

If your private key submission returns an error, the most common reasons are your private key is malformed or the system does not recognize the encryption method used for your key.

PRIVATE KEY CONSISTENCY

If you receive an error related to the private key, you can use OpenSSL to check its consistency:

```
openssl rsa -check -in <private_key>.pem
```

This command verifies that the key is intact, the passphrase is correct, and the file contains a valid RSA private key.

DECRYPTING A PRIVATE KEY

If the system does not recognize the encryption technology used for your private key, decrypt the key. Upload the unencrypted version of the key with your certificate bundle. You can use OpenSSL to decrypt a private key:

```
openssl rsa -in <private_key>.pem -out <decrypted_private_key>.pem
```

Using the Console

TO EDIT WAF SETTINGS

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to view settings for. The WAF Policy overview appears.
3. Click **Settings**.
4. Click **Edit**.

5. In the **Edit Settings** dialog box, enter the following:
    - **WAF Origin:** Select the name and IP address of the origin.
    - **Enable HTTPS Support:** When enabled, all communications between the browser and web app are encrypted. Enter the following information:
        - **SSL Certificate:** Drag and drop, select, or paste a valid SSL certificate in PEM format. You must also include intermediate certificates (the website certificate must be first). The following is an example:

```
-----BEGIN CERTIFICATE-----
<Base64_encoded_certificate>
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
<Intermediate_Base64_encoded_certificate>
-----END CERTIFICATE-----
```

        - **Private Key:** Drag and drop, select, or paste a valid private key in PEM format in this field. The private key cannot be protected by a passphrase. The following is an example:

```
-----BEGIN PRIVATE KEY-----
<Base64_encoded_private_key>
-----END PRIVATE KEY-----
```

        - **Self Signed Certificate:** Enable this field when using a self-signed certificate to show an SSL warning in the browser.
        - **HTTP to HTTPS Redirect:** When enabled, all HTTP traffic is automatically redirected to HTTPS.
6. Click **Save**.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

- CreateCertificate
- GetCertificate

- DeleteCertificate

Logs

Logs displays log activity and the details of each logged event within a specified time frame. Logs enable you to understand what rules and countermeasures are triggered by requests and are used as a basis to move request handling into block mode. Logs can come from Access Control, Protection Rules, or Bot events.



Note

If you have concerns over General Data Protection Regulation (GDPR) requirements, Logs can be disabled for the WAF service. You can use My Oracle Support to file a service request to disable Logs.

Using the Console

To view Logs

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Edge Services** and click **WAF Policies**.
2. Click the name of the WAF Policy you want to view logs for. The WAF Policy overview appears.
3. Click **Logs**. Logs for the WAF policy appear.
4. To help find a log, you can use the following filter options:
    - To view alerting activity data for a specific time range, enter a **Start Date**, **Start Time**, **End Date**, or **End Time**.
    - To view logs for a URL, enter a **Request URL**.
    - To view logs for a client IP address, select an address from the **Client IP Address** drop-down menu.

- To view logs for a country, select a country from the **Country Name** drop-down list.
- To find a type of action, select an **Action** check box.
- To find a log type, select a **Log Type** check box.

5. Click the plus sign next to the Alert Type you want to view.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

- ListWafLogs

CHAPTER 11 Email Delivery

This chapter explains how to send large volume email.

Overview of the Email Delivery Service

Oracle Cloud Infrastructure Email Delivery is an email sending service that provides a fast and reliable managed solution for sending high-volume emails that need to reach your recipients' inbox. Email Delivery provides the tools necessary to send application-generated email for mission-critical communications such as receipts, fraud detection alerts, multi-factor identity verification, and password resets.

Oracle Cloud Infrastructure's Email Deliverability team manages the platform using key deliverability metrics to ensure the best sending reputation possible for your emails.

The following items are provided to you when you send email using the Email Delivery service:

- Unique mailbox provider SMTP configurations on our Mail Transfer Agents (MTA)
- Bounce collection
- User complaint collection
- Email authentication standards
- Deliverability performance

Email Delivery Service Components

Email Delivery uses the components described in this section.

APPROVED SENDERS

An Approved Sender is a resource that equates to the "From" address. An approved sender is associated with a compartment and only exists in the region where the approved sender was configured. If you need to have the same approved sender in another region,

it must be created in the other region. For example, if you create an approved sender in the us-phoenix-1 region, you cannot send email through the us-ashburn-1 region.

SUPPRESSION LIST

The Suppression List is included on your Email Delivery console user interface and from the API. Email Delivery automatically adds email addresses with bounce codes showing permanent failures or user complaints to the suppression list to protect your sender reputation. Email Delivery will not send any messages to these recipients in the future.

Reasons for suppression currently include:

- Complaints
- Hard bounces
- Repetitive soft bounces
- Manual entries
- List-unsubscribe requests

SPF AUTHENTICATION

Sender Policy Framework (SPF) is used by email receivers to detect email spoofing. Using SPF, an email receiver can check if the Internet Protocol (IP) is explicitly authorized to send for that domain.

SPF is implemented by publishing a special TXT record to a domain's DNS records. The TXT record declares which hosts are allowed to send mail on behalf of this domain.

Receiving mail servers check the SPF records of sending domains to verify that the email's source IP address is authorized to send from that domain. Without SPF, a spam or phishing email can be “spoofed” to appear that the email comes from a legitimate domain. Domains that implement SPF are much more likely to block emails attempting to spoof your domain.

For an overview of how SPF works, see Sender Policy Framework. For details on SPF record syntax, see SPF Record Syntax.

Regions and Availability Domains

Email Delivery is available in the us-phoenix-1 and us-ashburn-1 regions. For more information, see Regions and Availability Domains.

The sending application is not required to be located in the region where email is sent. For example, if your sending application is located in a region where Email Delivery is not currently available, you would configure email from one of the regions where it is available. In the Console, change your region to us-phoenix-1 or us-ashburn-1 and create an approved sender. When creating SMTP credentials, any region can be used, as identities are global assets. Configure your application to send email to the region where you created the approved sender (us-phoenix-1 or us-ashburn-1 endpoint) using the SMTP credentials.

When Email Delivery is available in more regions, you can configure Email Delivery in the same region as the sending application to improve performance.

Ways to Access Oracle Cloud Infrastructure

You can access Oracle Cloud Infrastructure using the Console (a browser-based interface) or the REST API. Instructions for the Console and API are included in topics throughout this guide. For a list of available SDKs, see SDKs and Other Tools.

To access the Console, you must use a supported browser. You can use the Console link at the top of this page to go to the sign-in page. You are prompted to enter your cloud tenant, your user name, and your password. For general information about using the API, see About the API.

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see

Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

Email Delivery supports the following authentication types for control plane operations (management endpoint):

- **Instance Authorization:** The IAM service feature that enables instances to be authorized actors (or principals) to perform actions on service resources. Each compute instance has its own identity, and it authenticates using the certificates that are added to it. These certificates are automatically created, assigned to instances and rotated, preventing the need for you to distribute credentials to your hosts and rotate them.
- **Cross-Tenancy:** Cross-tenancy authorization allows customers to share resources between tenancies. To authorize a cross-tenancy request, the request must be endorsed by the requester's tenancy and permitted by the target tenancy.
- **Federated:** Federated authentication enables an administrator to configure a relationship between an identity provider and a service provider. When you federate Oracle Cloud Infrastructure with an identity provider, you manage users and groups in the identity provider. You manage authorization in Oracle Cloud Infrastructure's IAM service. Oracle Cloud Infrastructure tenancies are federated with Oracle Identity Cloud Service by default.



Note

Instance authorization, cross-tenancy, and federated authentication types do not apply to SMTP email sending. An approved sender and SMTP credentials are required and must be associated with the same tenancy for SMTP email sending.

SMTP Authentication and Connection Endpoints

Email Delivery only supports the AUTH PLAIN command when using SMTP authentication. If the sending application is not flexible with the AUTH command, an SMTP proxy/relay can be used. For more information about the AUTH command, see AUTH Command and its Mechanisms.

Use the following regional endpoints for establishing SMTP connections for sending.

- us-phoenix-1: smtp.us-phoenix-1.oraclecloud.com
- us-ashburn-1: smtp.us-ashburn-1.oraclecloud.com

Email Delivery Service Capabilities and Limits

See Service Limits for a list of applicable limits and instructions for requesting a limit increase.

Customers that sign up for a free Oracle Cloud trial are limited to:

- A volume of 200 emails a day.
- Five approved senders.
- Each user is limited to a maximum of two SMTP credentials.
- Sending rates are limited to ten emails per minute.
- Inline attachments.

Enterprise accounts are limited to:

- A volume of 50,000 emails a day.
- 10,000 approved senders.
- Sending rates are limited to 18,000 emails per minute.
- Inline attachments.



Note

The Email Delivery platform supports higher volumes. Limits are set as a safeguard for our customers' reputation. To file a service request to increase the email sending limit, open the navigation menu. Under **Governance and Administration**, go to **Service Limits**. Click **Request a service limit increase**.



Note

Currently, Email Delivery supports messages up to 2 MB, inclusive of message headers, body, and attachments. This is not a limit set per tenant. Larger messages sizes will be available in the future.

Required IAM Service Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

If you're new to policies, see Getting Started with Policies and Common Policies. For more details about policies for Email Delivery, see Details for the Email Service.

Permissions are required for managing and using approved senders and the suppression list. For example:

- To enable all operations on approved senders for a specific user group:

```
Allow group <Your Group Name> to manage approved-senders in tenancy
```

- To enable all operations on suppressions for a specific user group:

```
Allow group <Your Group Name> to manage suppressions in tenancy
```

Tagging Resources

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

Email Delivery supports applying tags to approved senders.

Integration with Oracle Cloud Infrastructure Services

Email Delivery audits the following events:

- Creating a sender (CreateSender)
- Deleting a sender (DeleteSender)
- Retrieving details about a sender (ListSenders)

To view logs for events in the Email Delivery service, your user must be in a group with the ability to view all of the Audit event logs in the tenancy. For more information, see Viewing Audit Log Events.

Getting Started with Email Delivery

The Oracle Cloud Infrastructure enables you to set up the Email Delivery service within the Console.

To begin sending email with Email Delivery, complete the following steps:

1. Generate SMTP credentials for a user.
2. Create an approved sender.

3. Configure SPF.
4. Configure the SMTP connection.

Generate SMTP Credentials for a User

Simple Mail Transfer Protocol (SMTP) credentials are necessary to send email through Email Delivery. Each user is limited to a maximum of two SMTP credentials. If more than two are required, SMTP credentials must be generated on other existing users or more users must be created.

A security best practice is to generate SMTP credentials for a new user instead of your Console user that already has permissions assigned to it. For detailed instructions on creating a user, see Adding Users. The new user must be assigned to a group with permissions to manage `approved-senders` and `suppressions`. For example:

```
Allow group <group name> to use approved-senders in compartment <compartment name>
```

Using the Console

To generate SMTP credentials for a user

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Users**. Locate the user in the list that has permissions to manage email, and then click the user's name to view the details.

   Tip

   If your user does not have permissions to view or create users, you can create SMTP credentials under your user. Open the **User** menu () and click **User Settings**.

2. Click **SMTP Credentials**.

3. Click **Generate SMTP Credentials**.
4. Enter a **Description** of the SMTP Credentials in the dialog box.
5. Click **Generate SMTP Credentials**. A user name and password is displayed.
6. Copy the user name and password for your records and click **Close**.

Managing Approved Senders

You must set up an approved sender for all "From:" addresses sending mail via Oracle Cloud Infrastructure or mail will be rejected. An approved sender is associated with a compartment and only exists in the region where the approved sender was configured. That is, if you create an approved sender in the us-phoenix-1 region, you cannot send email through the us-ashburn-1 region.

Approved senders should not be created in the root compartment. If approved senders exist in the root compartment, you are required to create a policy to manage approved senders in the entire tenant. Creating approved senders in a compartment other than the root allows the policy to be specific to that compartment.

Using the Console

To create an approved sender

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Email Delivery** and click **Email Approved Senders**. Ensure that you are in the correct compartment. Your user must be in a group with permissions to manage `approved-senders` in this compartment.
2. Click **Create Approved Sender** within the **Approved Senders** view.
3. Enter the email address you want to list as an approved sender in the **Create Approved Sender** dialog box.

Tags: Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

4. Click **Create Approved Sender**. The email address is added to your Approved Senders list.



Tip

Approved senders are unique to tenancies. If an attempt is made to create a duplicate approved sender within a tenancy, the service will return a 409 Conflict error.

To delete an approved sender

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Email Delivery** and click **Email Approved Senders**.
2. Find the approved sender you're interested in, click the Actions icon (three dots), and then click **Delete**.
3. In the confirmation dialog box, click **Confirm**. The email address is removed from the Approved Senders list.

To manage tags for an approved sender

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Email Delivery** and click **Email Approved Senders**.
2. Find the approved sender you're interested in, click the Actions icon (three dots), and then click **View Tags** to view or edit existing tags. Or click **Apply tag(s)** to add new

ones.

For more information, see Resource Tags.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operations to manage your approved senders:

- CreateSender
- GetSender
- ListSenders
- DeleteSender

Configure SPF

The Approved Senders section within the Console provides validation of an SPF record for each of your approved senders.

Using the Console

To configure SPF

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Email Delivery** and click **Email Approved Senders**.
2. Select the checkbox for the approved sender you want to view SPF details for and click **View SPF**.



Tip

You can search for an approved sender by using the Search field. Addresses can be sorted alphanumerically or by creation date in ascending or descending order.

3. The Manage SPF dialog box appears indicating whether an SPF record for the approved sender exists.
    - If your domain does not currently have an SPF record, the information necessary to add an SPF record in your DNS setup is displayed. See Managing DNS Service Zones for instructions on adding a zone record in Oracle Cloud Infrastructure. If your DNS setup resides with another provider, please reference their documentation for adding a TXT record to your domain.
        - In your DNS setup, create a TXT record and paste the following information from the dialog box into the record: `v=spf1 include:spf.oracleemaildelivery.com -all`
    - If your domain currently has an SPF record, add the following information to the record to add Oracle Cloud Infrastructure Email Delivery: `include:spf.oracleemaildelivery.com`

Configure SMTP Connection

Set up and test your SMTP connection using an SMTP library or product such as Postfix or SendMail to send email through Oracle Cloud Infrastructure Email Delivery.

To access SMTP sending information to configure the connection in your system, open the navigation menu. Under **Solutions, Platform and Edge**, go to **Email Delivery** and click **Email Configuration**. The following information is displayed:

- **Manage SMTP Credentials:** Access your user credentials. Use the SMTP user credentials (in plain text) when validating your connection.
- **Server Name:** The Email Delivery service hostname.
- **Port:** Email Delivery supports TLS on port 25 or 587.
- **Use Transport Layer Security (TLS):** This field indicates if TLS, the standard means of performing encryption in transit for email, is being used. Customers must encrypt email while it is in transit to the Oracle Cloud Infrastructure Email Delivery service. Encrypted emails are protected from being read during transit.



Important

Java applications (including JavaMail) must be updated to the latest version to ensure the latest protocols, ciphers, and security patches are in compliance with Oracle's supported security policies and ciphers.

SMTP Connection Endpoints

Use the following regional endpoints for establishing SMTP connections for sending.

- PHX: smtp.us-phoenix-1.oraclecloud.com
- IAD: smtp.us-ashburn-1.oraclecloud.com

TLS Requirements

Oracle maintains strict security policies and only accepts email traffic using Transport Layer Security (TLS). Use of TLS 1.2 is mandatory to send email using Oracle Cloud Infrastructure.

The approved TLS 1.2 ciphers are:

- TLS_DHE_RSA_WITH_AES_256_CBC_SHA
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA256
- TLS_DHE_DSS_WITH_AES_256_CBC_SHA256
- TLS_RSA_WITH_AES_256_CBC_SHA
- TLS_RSA_WITH_AES_256_CBC_SHA256
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA
- TLS_DHE_DSS_WITH_AES_128_CBC_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256

Managing the Suppression List

Manually add an email address to the suppression list to prevent it from being part of your sending list.

Users are required to have correct permissions to manage the suppression list. Currently, identity policies for suppression must be at the tenant level (not at the compartment level). The following is an example of the permission policy statement.

```
Allow group <group name> to manage suppressions in tenancy
```

Suppressions are stored at the tenancy level. Therefore any request requiring a `compartmentId` must provide the `tenancyId` as the `compartmentId`. For example:

```
Allow group <ordinary users> to inspect approved-senders in tenancy
Allow group <power users> to read approved-senders in tenancy
Allow group <sender admins> to manage approved-senders in tenancy
Allow user <mail user> to use approved-senders in tenancy where target.approved-sender.senderId =
<senderId>

Allow group <ordinary users> to inspect suppressions in tenancy
Allow group <power users> to read suppressions in tenancy
```

```
Allow group <sender admins> to manage suppressions in tenancy
```

Using the Console

To manually add an email address to the suppression list

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Email Delivery** and click **Email Suppression List**.
2. Click **Add Suppression**.
3. In the Add Suppression dialog box, enter the email address.
4. Click **Add**. The email address is added to the Suppression List.

To delete an email address from the suppression list

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Email Delivery** and click **Email Suppression List**.
2. Select the checkbox for the email address you want to delete and then click **Delete**.



Tip

You can search for an email address by using the Search field. Addresses can be sorted alphanumerically or by creation date in ascending or descending order.

3. In the confirmation dialog box, click **OK**. The email address is removed from the Suppression List.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operations to manage your suppressions:

- CreateSuppression
- GetSuppression
- ListSuppressions
- DeleteSuppression

Deliverability Best Practices

Deliverability Best Practices help you to learn and manage the habits that affect your sending reputation. These six recommendations can help lower your email bounce rate, stay off blacklists, lower your complaint rate, and improve your email sender reputation.

Implement an Opt-in Process

An opt-in process is a method for your users to subscribe to your mailing list, which gives you permission to send messages. Only send messages to subscribers who have opted-in to your mailing list. There are two types of opt-in procedures.

- **Single opt-in (unconfirmed):** A user provides their email address and gives permission to receive relevant messages. Once the address is provided, messages can be sent without confirming the email address belongs to the user who provided it.
- **Double opt-in (confirmed):** A user provides their email address, but before the first mailing, a confirmation email is sent to the account owner. The email requires action from the account owner to confirm that future messages are wanted. An account can be verified by having the owner click a link for reply to the email. The confirmation email ensures that the address was not added to a third-party mailing list without consent.

Purge Unengaged Users

Remove unengaged users by implementing a process. If a recipient is not engaging with your mail by either opening or clicking the email, this might be an indication that the email account is not in use or that the recipient is no longer interested in your content. If the recipient does not use the email account, eventually the mailbox provider terminates the account or transforms the account into a spam trap. Remove recipients who have not engaged with your email in a time frame defined by your business model. Purging unengaged users helps your deliverability by increasing your user engagement rate.

Review Your Subscriber List

When reviewing your subscriber list, keep these things in mind:

- Eliminate duplicate addresses before sending. If addresses that do not exist are mailed to multiple times, your hard bounce rate could be inflated.
- Ensure that a previous suppression list (possibly from another email service provider) was not accidentally included.
- Verify that subscribers have opted-in. Do not send to an old list that you found.
- Restrict users from uploading their email client's contact list in a "select all" fashion. Forcing users to select addresses individually prevents users from accidentally including potentially out of date or expired addresses.

Evaluate Your Sending Frequency

Sending too many emails in a short time might aggravate recipients, causing the recipients to mark your messages as spam. This is called list fatigue. Ensure that your message cadence aligns with the expected frequency of your content. Reducing frequency might reduce spam complaints. Ensure that your content is relevant to your subscribers. Keep your email messages consistent to your audience. A person who subscribed to a list for coupon updates might not want regular emails about auto loan finance rates. These unexpected messages are likely to be marked as spam, which decreases your sender reputation.

Easily Accessible Unsubscribe URL

Unsubscribing helps your inbox success by sending only to recipients that engage by opening or clicking. When people complain, your sending reputation is harmed. Make it easy for recipients to be removed from the list. Do not hide the unsubscribe URL at the bottom of the message. A small percentage of users scroll to the bottom of the email and search for a small URL. Most users mark the email as spam.

Canadian Anti-Spam Law (CASL) Guide

Canada's Anti-Spam Law (CASL) is one of the best guides to ensuring your compliance with the law, users' desire, and the intended filtering that most mailbox providers use. If you are a Canadian email sender or you send email to Canadian residents, you must comply with CASL. The following information is intended to help provide you with some guidance for complying with CASL. This article does not constitute legal advice, nor is it intended supplement or otherwise affect your rights or obligations under your service agreement with Oracle, including your obligations under Oracle's Acceptable Use Policy. If you have questions about CASL or the legality of your sending practices, we encourage you to speak with an attorney who specializes in that subject matter.

What is covered by CASL?

CASL and its related regulations apply to any "commercial electronic message" sent from or to Canadian computers and devices in Canada. Electronic messages that are merely routed through Canadian computer systems are not subject to CASL.

A "commercial electronic message" is any message that:

- Is in an electronic format, including emails, instant messages, text messages, and some social media communications.
- Is sent to an electronic address, including email addresses, instant message accounts, phone accounts, and social media accounts; and
- Contains a message encouraging recipients to take part in some type of commercial activity, including the promotion of products, services, people/personas, companies, or organizations.

Are there any types of messages that are exempt from CASL?

These types of electronic messages are exempt from CASL for various reasons.

- Messages to family or a person with established personal relationship.
- Messages to an employee, consultant, or person associated with your business.
- Responses to a current customer, or someone who has inquired in the last six months.
- Messages that will be opened or accessed in a foreign country, including the U.S., China, and most of Europe.
- Messages sent on behalf of a charity or political organization for the purposes of raising funds or soliciting contributions.
- Messages attempting to enforce a legal right or court order.
- Messages that provide warranty, recall, safety, or security information about a product or service purchased by the recipient.
- Messages that provide information about a purchase, subscription, membership, account, loan, or other ongoing relationship, including delivery of product updates or upgrades.
- A single message to a recipient without an existing relationship based on a referral. The full name of the referring person must be disclosed in the message. The referrer might be family or have another relationship with the person to whom you are sending.

If your message does not meet one of these criteria, consent is required under CASL. Not all of the previous messages listed are permitted under the Oracle Cloud Hosting and Delivery Policy.

What is "express consent"?

Under CASL, "express consent" means a written or oral agreement to receive specific types of messages. For example, "You want to receive monthly newsletters and weekly discount notifications from Oracle".

Express consent is only valid if your request for consent clearly and simply describes the following information:

- Your purpose in obtaining consent.
- A description of messages you will be sending.
- The name and contact information (physical mailing address and telephone number, email address, or website URL) of the requestor.
- A statement that the recipient can unsubscribe at any time.

The requestor can be you or someone for whom you are asking. If you are requesting consent on behalf of a client, the name and contact information of the client must be included with the consent request.

What is "implied consent"?

Under CASL, you can only obtain implied consent when certain circumstances exist, including when:

- A recipient has purchased a product, service or made another business deal, contract, or membership with your organization in the last 24 months.
- You are a registered charity or political organization, and the recipient has made a donation or gift, has volunteered, or attended a meeting organized by you.
- A professional message is sent to someone whose email address was given to you, or is conspicuously published, and who has not published or told you that unsolicited messages are not wanted.

What type of consent is required?

After July 1, 2017, you can only send to recipients with express consent or whose implied consent is valid under CASL.

Some additional requirements

In addition to understanding what qualifies as CASL-regulated message, and what type of consent is needed, there are a few other details to keep in mind.

- Retention of a record of consent confirmations is required.
- When requesting consent, checkboxes cannot be pre-filled to suggest consent. Each subscriber must check the box themselves for consent to be valid.

- All messages sent must include the following:
  - your name
  - the person on whose behalf you are sending (if any)
  - your physical mailing address and telephone number
  - your email address or website URL
- All messages sent after consent must also include an unsubscribe mechanism, and unsubscribes must be processed within ten days.

Where can I find more information on CASL?

The full text of the law can be found on the website for the Canadian Justice Department. The Canadian Radio and Telecommunications Commission has also set up an FAQ page and some guidelines for obtaining consent. If you have any questions, we encourage you to contact an attorney who is familiar with the law.

Oracle Cloud Hosting and Delivery Policy

Often, the Oracle Cloud Hosting and Delivery Policy is more stringent than CASL requirements. It is important that you review Oracle policies before using the service.

Troubleshooting Undelivered Emails

The following issues can cause an email to be undelivered:

- The recipient is on the Suppression List.
- An authentication failure or an issue with the format of the email message occurred. For example, if the SMTP "From" address is not the same as the "From" address in the email body, the email is rejected. The addresses must match and be an Approved Sender. Refer to your sending application's logs to review any issues.

If you are unable to resolve the issue, you can go to My Oracle Support and create a service request. See Creating a Service Request for more information.

CHAPTER 12 File Storage

This chapter explains how to create file systems, how to manage them, and how to mount them to write files.

Overview of File Storage

Oracle Cloud Infrastructure File Storage service provides a durable, scalable, secure, enterprise-grade network file system. You can connect to a File Storage service file system from any bare metal, virtual machine, or container instance in your Virtual Cloud Network (VCN). You can also access a file system from outside the VCN using Oracle Cloud Infrastructure FastConnect and Internet Protocol security (IPSec) virtual private network (VPN).

Large Compute clusters of thousands of instances can use the File Storage service for high-performance shared storage. Storage provisioning is fully managed and automatic as your use scales from a single byte to exabytes without upfront provisioning. You have redundant storage for resilient data protection.

The File Storage service supports the Network File System version 3.0 (NFSv3) protocol. The service supports the Network Lock Manager (NLM) protocol for file locking functionality.

Use the File Storage service when your application or workload includes big data and analytics, media processing, or content management, and you require Portable Operating System Interface (POSIX)-compliant file system access semantics and concurrently accessible storage. The File Storage service is designed to meet the needs of applications and users that need an enterprise file system across a wide range of use cases, including the following:

- **General Purpose File Storage:** Access to an unlimited pool of file systems to manage growth of structured and unstructured data.
- **Big Data and Analytics:** Run analytic workloads and use shared file systems to store persistent data.

- **Lift and Shift of Enterprise Applications:** Migrate existing Oracle applications that need NFS storage, such as Oracle E-Business Suite and PeopleSoft.
- **Databases and Transactional Applications:** Run test and development workloads with Oracle, MySQL, or other databases.
- **Backups, Business Continuity, and Disaster Recovery:** Host a secondary copy of relevant file systems from on premises to the cloud for backup and disaster recovery purposes.
- **MicroServices and Docker:** Deliver stateful persistence for containers. Easily scale as your container-based environments grow.



Tip

Watch a video introduction to the service and its capabilities.

File Systems Concepts

Using the File Storage service requires an understanding of the following concepts, including some that pertain to Oracle Cloud Infrastructure Networking:

MOUNT TARGET

An NFS endpoint that lives in a subnet of your choice and is highly available. The mount target provides the IP address or DNS name that is used in the mount command when connecting NFS clients to a file system. A single mount target can export many file systems. By default, you can create two mount targets per account per availability domain, but you can request an increase. See Service Limits for a list of applicable limits and instructions for requesting a limit increase. See Managing Mount Targets for more information about working with this resource.

Export

Exports control how NFS clients access file systems when they connect to a mount target. File systems are exported (made available) through mount targets. Each mount target maintains an export set which contains one or many exports. A file system must have at least one export in one mount target in order for instances to mount the file system. The information used by an export includes the file system OCID, mount target OCID, export set OCID, export path, and client export options. For more information, see Managing Mount Targets.

Export Set

Collection of one or more exports that control what file systems the mount target exports using NFSv3 protocol and how those file systems are found using the NFS mount protocol. Each mount target has an export set. Each file system associated with the mount target has at least one export in the export set.

Export Path

A path that is specified when an export is created. It uniquely identifies the file system within the mount target, letting you associate up to 100 file systems to a single mount target. This path is unrelated to any path within the file system itself, or the client mount point path.

The File Storage service adds an export that pairs the file system's Oracle Cloud Identifier (OCID) and path.

See Paths in File Systems for more information.

Export Options

NFS export options are a set of parameters within the export that specify the level of access granted to NFS clients when they connect to a mount target. An NFS export options entry within an export defines access for a single IP address or CIDR block range. For more information, see Working with NFS Export Options.

Virtual Cloud Network (VCN)

A private network that you set up in the Oracle data centers, with firewall rules and specific types of communication gateways that you can choose to use. A VCN covers a single, contiguous IPv4 CIDR block of your choice. For more information about VCNs, see VCNs and Subnets in the Oracle Cloud Infrastructure Networking documentation.

You can set up a service gateway and give your VCN private access to the File Storage service. A service gateway can be used only by resources in the gateway's own VCN. Traffic to the service will not travel through the internet. When creating the service gateway, enable the service label called **All <region> Services in Oracle Services Network**. It includes the File Storage service. Be sure to update route tables for any subnets that need to access File Storage through the service gateway.

For more information and detailed instructions, see Setting Up a Service Gateway in the Console

Subnets

Subdivisions you define in a VCN (for example, 10.0.0.0/24 and 10.0.1.0/24). Subnets contain virtual network interface cards (VNICs), which attach to instances. A subnet can span a region or exist in a single availability domain . A subnet consists of a contiguous range of IP addresses that do not overlap with other subnets in the VCN. For each subnet, you specify the routing rules and security lists that apply to it. For more information about subnets, see VCNs and Subnets in the Oracle Cloud Infrastructure Networking documentation.

SECURITY LISTS

Virtual firewall rules for your VCN. Your VCN comes with a default security list, and you can add more. These security lists provide ingress and egress rules that specify the types of traffic allowed in and out of the instances. You can choose whether a given rule is stateful or stateless. Security list rules must be set up so that clients can connect to file system mount targets. For more information about how security lists work in Oracle Cloud Infrastructure, see Security Lists in the Networking documentation. For information about setting up specific security list rules required for mount target traffic, see Configuring VCN Security List Rules for File Storage. About Security explains how security lists interact with other types of security in your file system.

SNAPSHOTS

Snapshots provide a consistent, point-in-time view of your file system, and you can take as many snapshots as you need. You pay only for the storage used by your data and metadata, including storage capacity used by snapshots. Each snapshot reflects only data that changed from the previous snapshot. For more information, see Managing Snapshots.

Encryption

All files are encrypted at rest by default.

Data Transfers

FastConnect offers you the ability to accelerate data transfers. You can leverage the integration between FastConnect and the File Storage service to perform initial data migration, workflow data transfers for large files, and disaster recovery scenarios between two regions, among other things.

File Storage Space Allocation

The File Storage service allocates space in blocks of variable size in a way that is fine-tuned to minimize total customer cost and optimize performance for modern workloads. The minimum block size used is 8192 bytes. For example, if you create a 1-byte file, we allocate

8192 bytes. We use larger blocks to store larger files. This method of allocation might cause a different block count for files than expected after they are copied from another storage device to your Oracle Cloud Infrastructure file system.

How File Storage Permissions Work

File Storage service resources include file systems, mount targets, and export sets. The AUTH_UNIX style of authentication and permission checking is supported for remote NFS client requests. You use Oracle Cloud Infrastructure Identity and Access Management (IAM) policy language to define access to Oracle Cloud Infrastructure resources. You can consider exports and snapshots subsidiary resources of export sets and file systems, respectively. As such, they do not need their own permissions. Related resources include Oracle Cloud Infrastructure Compute instances and Oracle Cloud Infrastructure Networking virtual cloud networks (VCNs).

Oracle Cloud Infrastructure users require resource permissions to create, delete, and manage resources. Without the appropriate IAM permissions, you cannot export a file system through a mount target. Until a file system has been exported, Compute instances cannot mount it. For more information about creating an IAM policy, see Let users create, manage, and delete file systems.

If you have successfully exported a file system on a subnet, then you use Networking security lists to control traffic to and from the subnet and, therefore, the mount target. Security lists act as a virtual firewall, allowing only the network traffic you specify to and from the IP addresses and port ranges configured in your ingress and egress rules. The security list you create for the subnet lets hosts send and receive packets and mount the file system. If you have firewalls on individual instances, use FastConnect, or use a virtual private network (VPN), the settings for those might also impact security at the networking layer. For more information about creating a security list for the File Storage service, see Creating File Systems. See About Security for more information on how different types of security work together in your file system.

Regions and Availability Domains

You can use the File Storage service in all regions. For a list of supported regions, see Regions and Availability Domains.

You can share mount targets for file systems among local bare metal and virtual Compute resources within a region. When you create file systems and mount targets, you specify the availability domain they are created in. While it is possible to access mount targets from any AD in a region, for optimal performance, place File Storage resources in the same availability domain as the Compute instances that access them.

Subnets can be either AD-specific or regional. You can create File Storage resources in either type of subnet. Regional subnets allow Compute instances to connect to any mount target in the subnet regardless of AD, with no additional routing configuration. However, to minimize latency, place mount targets in the same AD as Compute instances just as you would in an AD-specific subnet. For more information, see About Regional Subnets.

Within an availability domain, the File Storage service uses synchronous replication and high availability failover to keep your data safe and available.

Resource Identifiers

Most types of Oracle Cloud Infrastructure resources have a unique, Oracle-assigned identifier called an Oracle Cloud ID (OCID). For information about the OCID format and other ways to identify your resources, see Resource Identifiers.

Ways to Access Oracle Cloud Infrastructure

You can access Oracle Cloud Infrastructure using the Console (a browser-based interface) or the REST API. Instructions for the Console and API are included in topics throughout this guide. For a list of available SDKs, see Software Development Kits and Command Line Interface.

To access the Console, you must use a supported browser. Oracle Cloud Infrastructure supports the latest desktop versions of Google Chrome, Microsoft Edge, Internet Explorer 11,

Safari, Firefox, and Firefox ESR. Note that private browsing mode is not supported for Firefox, Internet Explorer, or Edge. Mobile browsers are not supported.

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

Limits on Your File Storage Components

See Service Limits for a list of applicable limits and instructions for requesting a limit increase.

About Security

The File Storage service uses four different layers of security. Each layer has its own authorization entities and methods which are separate from the other layers.



Tip

Watch a video about security in File Storage.

The **Oracle Cloud Infrastructure (OCI) policy** layer uses policies to control what users can do within Oracle Cloud Infrastructure, such as creating instances, a VCN and its security rules, mount targets, and file systems.

The **Network security** layer controls which instance IP addresses or CIDR blocks can connect to a host file system. It uses VCN security list rules to allow or deny traffic to the mount target, and therefore access to any associated file system.

The **NFS export option** layer is a method of applying access control per-file system export based on source IP address that bridges the Network Security layer and the NFS v.3 Unix Security layer.

The **NFS v.3 Unix security** layer controls what users can do on the instance, such as installing applications, creating directories, mounting external file systems by a local mount point, and reading and writing files.

This security layer...	Uses these...	To control actions like...
Oracle Cloud Infrastructure (Oracle Cloud Infrastructure)	OCI Users and policies	Creating instances and VCNs. Creating, listing, and associating file systems and mount targets.
Network security	IP addresses, CIDR blocks, security lists	Connecting the client instance to the mount target.

This security layer...	Uses these...	To control actions like...
NFS export options	File system exports, IP addresses, Unix users	Privileged source port connection, reading and writing files, and limiting root user access on a per-file system basis.
NFS v.3 Unix security	Unix users, file mode bits	Mounting file systems, reading and writing files.

You create users and groups in Oracle Cloud Infrastructure. Then, you can use policies to specify which users and groups can create, access, or modify resources such as file systems, mount targets, and export options.

The network security layer allows you to use VCN security lists to block the appropriate ports from specific IP addresses and CIDR blocks and restrict host access. However, it's on an 'all or nothing' basis - the client either can or cannot access the mount target, and therefore all file systems associated with it. See Working with NFS Export Options to specify granular controls on a per-file system basis.

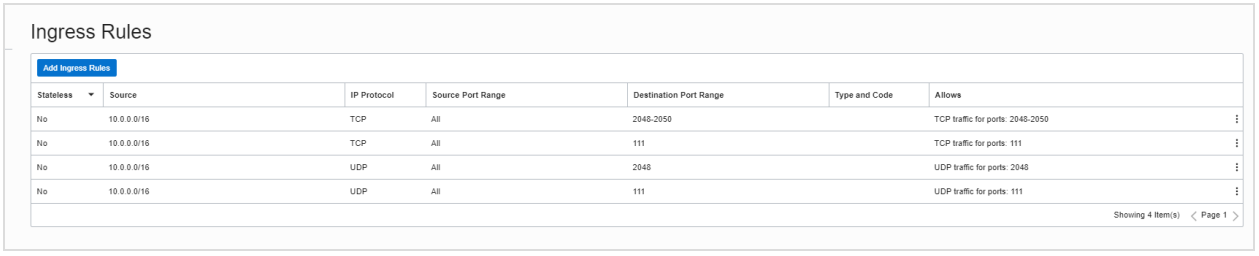
File Storage service supports the AUTH_UNIX style of authentication and permission checking for remote NFS client requests. When mounting file systems, we recommend that you use the `-nosuid` option. This option disables set-user-identifier or set-group-identifier bits. Remote users are prevented from gaining higher privileges using a `setuid` program. For more information, see Mounting File Systems.

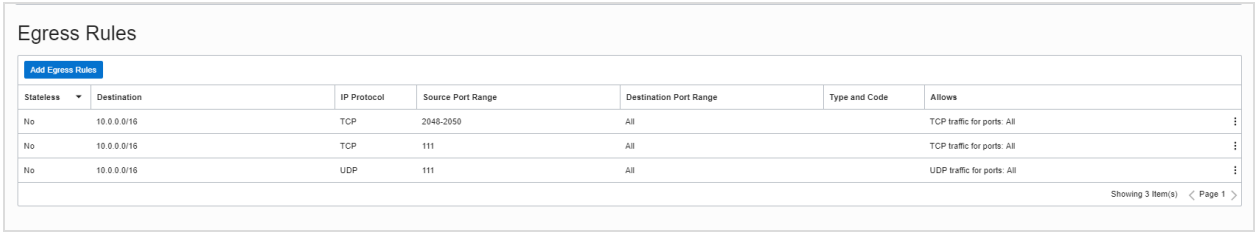
Remember that users in UNIX aren't the same as users in Oracle Cloud Infrastructure - they're not linked or associated in any way. The Oracle Cloud Infrastructure policy layer doesn't govern anything that happens inside the file system, the UNIX security layer does. Conversely, the UNIX security layer doesn't govern creating file systems or mount targets in Oracle Cloud Infrastructure.

NFS export options are a method of applying access control at the network security layer and the NFS v.3 Unix security layer. You can use NFS export options to limit access levels by IP addresses or CIDR blocks connecting to multiple file systems through exports of an associated mount target. Access can be restricted so that each client's file system is inaccessible and invisible to the other, allowing for managed hosted environment security. Moreover, you can set permissions for read-only, read/write, or root-squash for your file systems. See Working with NFS Export Options for more information.

Configuring VCN Security List Rules for File Storage

When you create a VCN, a default security list is also created. Rules in the security list are used to allow or deny traffic to a subnet. Before you can mount a file system, you must configure security list rules to allow traffic to the mount target subnet. File Storage requires stateful ingress to TCP ports 111, 2048, 2049, and 2050 and stateful ingress to UDP ports 111 and 2048. File storage also requires stateful egress from TCP ports 111, 2048, 2049, and 2050 and stateful egress from UDP port 111.





See Security Lists for more information about how security lists work in Oracle Cloud Infrastructure. See About Security for information about how security lists work with other types of security in File Storage.

Required IAM Service Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let network admins manage a cloud network covers management of all networking components, including security lists. See the Policy Reference for more information.

If you're new to policies, see Getting Started with Policies and Common Policies.

Using the Console

To configure security list rules for mount target traffic

Security list rules allow ingress and egress for the following:

- Open Network Computing Remote Procedure Call (ONC RPC) rpcbind utility protocol
- Network File System (NFS) protocol
- Network File System (MOUNT) protocol
- Network Lock Manager (NLM) protocol

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. In the **Scope** section, select the compartment that contains the subnet associated with your file system.
3. Click the name of the cloud network associated with your file system.

4. On the details page for the cloud network, in **Resources**, and then click **Security Lists**.
5. Click the name of the security list used by the subnet associated with your file system.
6. In **Resources**, click **Ingress Rules**.
7. Click **Add Ingress Rules** and add the following ingress rule allowing **TCP** traffic.
    - Specify that it's a stateful rule by leaving the check box clear. (For more information about stateful and stateless rules, see Stateful vs. Stateless Rules). By default, rules are stateful unless you specify otherwise.
    - To allow traffic from the subnet of the cloud network, click **Source Type**, choose **CIDR**, and then enter the CIDR block for the subnet.
    - Click **IP Protocol**, and then click **TCP**.
    - In **Source Port Range**, specify the range of ports that you want to allow traffic from. Alternatively, accept the default of **All** to allow traffic from any source port.

      Important

      We recommend that NFS clients be limited to reserved ports. To do this, set the **Source Port** range to **1-1023**. You can also set export options for a file system to require clients to connect from a privileged source port. For more information, see Working with NFS Export Options.
    - Click **Destination Port Range**, and then enter **2048-2050**.
8. Click **+ Additional Ingress Rule** and create a second stateful ingress rule allowing **TCP** traffic to a **Destination Port Range** of **111**.
9. Click **+ Additional Ingress Rule** and create a third stateful ingress rule allowing **UDP** traffic to a **Destination Port Range** of **2048**.

10. Click **+ Additional Ingress Rule** and create a fourth stateful ingress rule allowing **UDP** traffic to a **Destination Port Range** of **111**.
11. When you're done, click **Add Ingress Rules**.
12. Next, create the egress rules. In **Resources**, click **Egress Rules**.
13. Click **Add Egress Rules** and add the following egress rule allowing **TCP** traffic:
    - Specify that it's a stateful rule by leaving the check box clear.
    - Click **Destination Type**, choose **CIDR**, and then enter the CIDR block for the subnet.
    - Click **IP Protocol**, and then click **TCP**.
    - In **Source Port Range**, enter **2048-2050**.
    - In **Destination Port Range**, accept the default of **All** to allow traffic to any destination port.
14. Click **+ Additional Ingress Rule** and add a second stateful egress rule allowing **TCP** traffic from a **Source Port Range** of **111**.
15. Click **+ Additional Ingress Rule** and add a third stateful egress rule allowing **UDP** traffic from a **Source Port Range of 111**.
16. When you're done, click **Add Egress Rules**.

Next steps:

- Create a mount target and associated file system
- Mount a file system

Working with NFS Export Options

This topic describes the basic features of NFS export options, and how to control client access to your file system.

Overview

NFS export options enable you to create more granular access control than is possible using just security list rules to limit VCN access. You can use NFS export options to specify access levels for IP addresses or CIDR blocks connecting to file systems through exports in a mount target. Access can be restricted so that each client's file system is inaccessible and invisible to the other, providing better security controls in multi-tenant environments.

Using NFS export option access controls, you can limit clients' ability to connect to the file system and view or write data. For example, if you want to allow clients to consume but not update resources in your file system, you can set access to Read Only. You can also reduce client root access to your file systems and map specified User IDs (UIDs) and Group IDs (GIDs) to an anonymous UID/GID of your choice. For more information about how NFS export options work with other security layers, see About Security.



Tip

Watch a video about working with NFS export options in File Storage.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let users create, manage, and delete file systems allows users to manage NFS export options.

If you're new to policies, see Getting Started with Policies and Common Policies.

Exports

Exports control how NFS clients access file systems when they connect to a mount target. File systems are exported (made available) through mount targets. Each mount target maintains an export set which contains one or many exports. A file system may be exported through one or more mount targets. A file system must have at least one export in one mount target in order for instances to mount the file system. The information used by an export includes the file system OCID, mount target OCID, export set OCID, export path, and client export options. Typically, an export is created in a mount target when the file system is created. Thereafter, you can create additional exports for a file system in any mount target that resides in the same availability domain as the file system.

See To create an export for a file system for more information.

NFS Export Options

NFS export options are a set of parameters within the export that specify the level of access granted to NFS clients when they connect to a mount target. An NFS export options entry within an export defines access for a single IP address or CIDR block range.

Each separate client IP address or CIDR block you want to define access for needs a separate export options entry in the export. For example, if you want to set options for NFS client IP addresses 10.0.0.6, 10.0.08, and 10.0.0.10, you need to create three separate entries, one for each IP address.

File Storage service considers the listed order of each export options entry for the export. During an NFS request by a client, File Storage service applies the first set of options that matches the client Source IP address. Only the first set is applied; the rest are ignored.

For example, consider the following two export options entries specifying access for an export:

Entry 1: Source: 10.0.0.0/16, Access: Read Only

Entry 2: Source: 10.0.0.8, Access: Read/Write

In this case, clients who connect to the export from IP address 10.0.0.8 have Read Only access. The request Source IP address is contained in the CIDR block specified in the first entry, and File Storage Service applies the options in the first match.



Important

File systems can be associated with one or more exports, contained within one or more mount targets. If the client **source** IP address does not match any entry on the list for a single export, then that export is not visible to the client. However, the file system could be accessed through other exports on the same or other mount targets. **To completely deny client access to a file system, be sure that the client source IP address or CIDR block is not included in any export for any mount target associated with the file system.**

The following options can be set to control export access:

- **Source**: The IP address or CIDR block of a connecting NFS client.
- **Require Privileged Source Port (true/false):** This setting determines whether the NFS clients specified in **source** are required to connect from a privileged source port. Privileged ports are any port including 1-1023. On Unix-like systems, only the root user can open privileged ports. Setting this value to **true** disallows requests from unprivileged ports. The default for this setting is different depending on how the export is created. Creating an export without an explicit `ClientOption` array sets the `requirePrivilegedSourcePort` attribute of the client option to **false**. When you create a `ClientOption` array explicitly , `requirePrivilegedSourcePort` defaults to **true**. For example, creating an export in the Console using the default selections sets `requirePrivilegedSourcePort` to **false**. Creating an export in the API along with a `ClientOption` array sets `requirePrivilegedSourcePort` to **true**.



Important

When **Require Privileged Source Port** is set to **true**, you also have to follow these additional configuration steps:

1. When mounting the file system from a Unix-like system, include the `resvport` option in your mount command when mounting. For example:

   ```
   sudo mount -o resvport 10.x.x.x:/fs-export-path
   /mnt/yourmountpoint
   ```

   For more information, see Mounting File Systems From Unix-Style Instances.
2. When mounting the file system from a Windows system, be sure the **UseReserverdPort**s registry key value is set to **1**.
   For more information, see Mounting File Systems From Windows Instances.

- **Access (Read_Only, Read_Write):** This setting specifies the **source** NFS client access. If unspecified, defaults to **Read_Write**.
- **Identity Squash: (All, Root, None):** This setting determines whether the **source** clients accessing the file system have their User ID (UID) and Group ID (GID) remapped to **anonymousUid** and **anonymousGid**. If you choose **All**, all users and groups are remapped. If **Root**, only the root user UID/GID combination 0/0 is remapped. If **None**, no users are remapped. If unspecified, defaults to **None**.
- **anonymousUid:** This setting is used along with the **Identity Squash** option. When remapping users, you can use this setting to change the default anonymousUid of **65534** to any user ID of your choice.

- **anonymousGid**: This setting is used along with the **Identity Squash** option. When remapping groups, you can use this setting to change the default anonymousGid of **65534** to any group ID of your choice.

Typical Access Control Scenarios

When you create file system and export, the NFS export options for that file system are set to the following defaults, which allow full access for all NFS client source connections. These defaults must be changed if you want to restrict access:

- **Source:** 0.0.0.0/0 (All)
- **Require Privileged Source Port:** False
- **Access:** Read_Write
- **Identity Squash:** None

Scenario A: Control Host Based Access

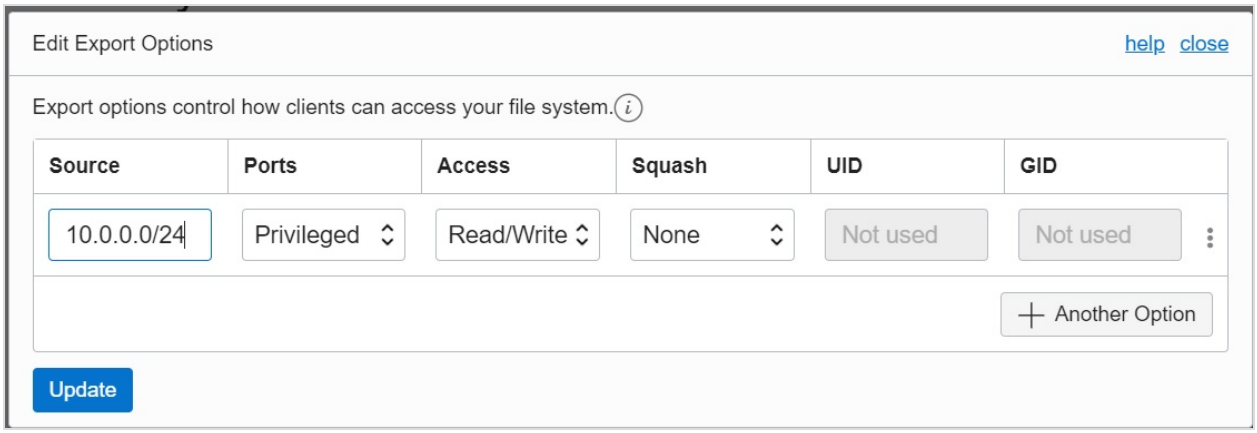
Provide a managed hosted environment for two clients. The clients share a mount target, but each has their own file system, and cannot access each other's data. For example:

- Client A, who is assigned to CIDR block 10.0.0.0/24, requires Read/Write access to file system A, but not file system B.
- Client B, who is assigned to CIDR block 10.1.1.0/24, requires Read/Write access to file system B, but not file system A.
- Client C, who is assigned to CIDR block 10.2.2.0/24, has no access of any kind to file system A or file system B.
- Both file systems A and B are associated to a single mount target, MT1. Each file system has an export contained in the export set of MT1.

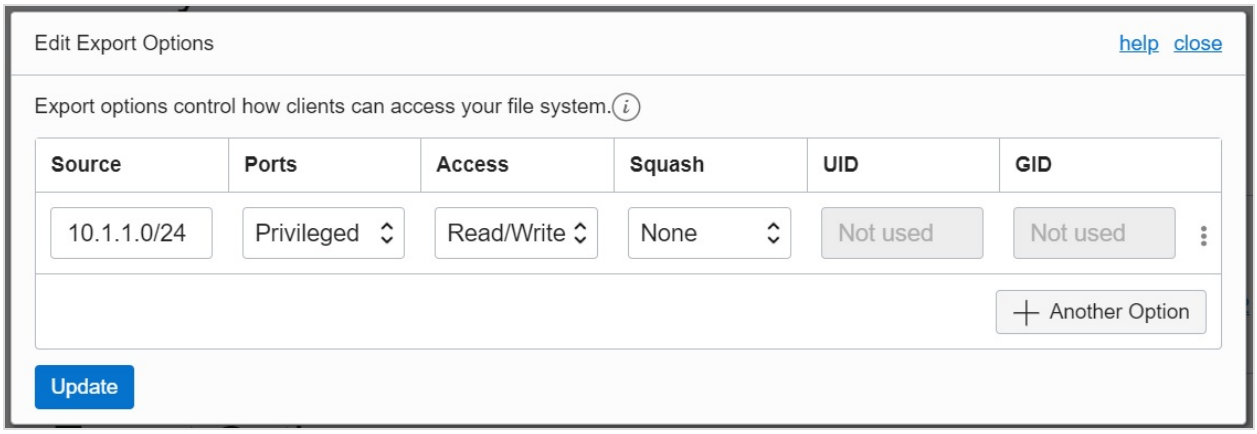
Since Client A and Client B access the mount target from different CIDR blocks, you can set the client options for both file system exports to allow access to only a single CIDR block. Client C is denied access by not including its IP address or CIDR block in the NFS export options for any export of either file system.

Console Example

Set the export options for file system A to allow Read/Write access only to Client A, who is assigned to CIDR block 10.0.0.0/24. Client B and Client C are not included in this CIDR block, and cannot access the file system.



Set the export options for file system B to allow Read/Write access only to Client B, who is assigned to CIDR block 10.1.1.0/24. Client A and Client C are not included in this CIDR block, and cannot access the file system.



CLI Example

Set the export options for file system A to allow Read_Write access only to Client A, who is assigned to CIDR block 10.0.0.0/24. Client B and Client C are not included in this CIDR block, and cannot access the file system.

```
oci fs export update --export-id <File_system_A_export_ID> --export-options '
[{"source":"10.0.0.0/24","require-privileged-source-port":"true","access":"READ_WRITE","identity-
squash":"NONE","anonymous-uid":"65534","anonymous-gid":"65534"}]'
```

Set the export options for file system B to allow Read_Write access only to Client B, who is assigned to CIDR block 10.1.1.0/24. Client A and Client C are not included in this CIDR block, and cannot access the file system.

```
oci fs export update --export-id <File_system_B_export_ID> --export-options '[{"source":"10.1.1.0/24
","require-privileged-source-port":"true","access":"READ_WRITE","identity-squash":"NONE","anonymous-
uid":"65534","anonymous-gid":"65534"}]'
```

API Example

Set the export options for file system A to allow READ_WRITE access only to Client A, who is assigned to CIDR block 10.0.0.0/24. Client B and Client C are not included in this CIDR block, and cannot access the file system.

```
PUT/<Current_API_Version>/exports/<File_System_A_export_OCID>
Host: filestorage.us-phoenix-1.oraclecloud.com
<authorization and other headers>
{
    "exportOptions": [
        {
          "source": "10.0.0.0/24",
          "requirePrivilegedSourcePort": true,
          "access": "READ_WRITE",
          "identitySquash": "NONE",
          "anonymousUid": 65534,
          "anonymousGid": 65534
        }
    ]
}
```

Set the export options for file system B to allow READ_WRITE access only to Client B, who is assigned to CIDR block 10.1.1.0/24. Client A and Client C are not included in this CIDR block, and cannot access the file system.

```
PUT/<Current_API_Version>/exports/<File_System_B_export_OCID>
Host: filestorage.us-phoenix-1.oraclecloud.com
<authorization and other headers>
{
    "exportOptions": [
        {
          "source": "10.1.1.0/24",
          "requirePrivilegedSourcePort": true,
          "access": "READ_WRITE",
          "identitySquash": "NONE",
          "anonymousUid": 65534,
          "anonymousGid": 65534
        }
    ]
}
```

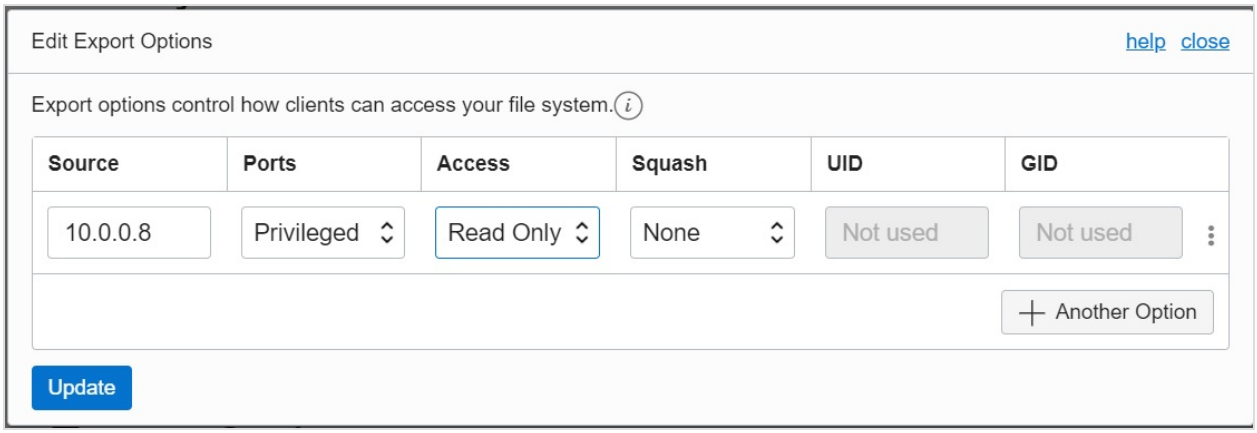
Scenario B: Limit the Ability to Write Data

Provide data to customers for consumption, but don't allow them to update the data.

For example, you'd like to publish a set of resources in file system A for an application to consume, but not change. The application connects from IP address 10.0.0.8.

Console Example

Set the source IP address 10.0.0.8 to Read Only in the export for file system A:



CLI Example

Set the source IP address 10.0.0.8 to READ_ONLY in the export for file system A:

```
oci fs export update --export-id <File_System_A_export_OCID> --export-options '
[{"source":"10.0.0.8","require-privileged-source-port":"true","access":"READ_
ONLY","identitysquash":"NONE","anonymousuid":"65534","anonymousgid":"65534"}]'
```

API Example

Set the source IP address 10.0.0.8 to READ_ONLY in the export for file system A:

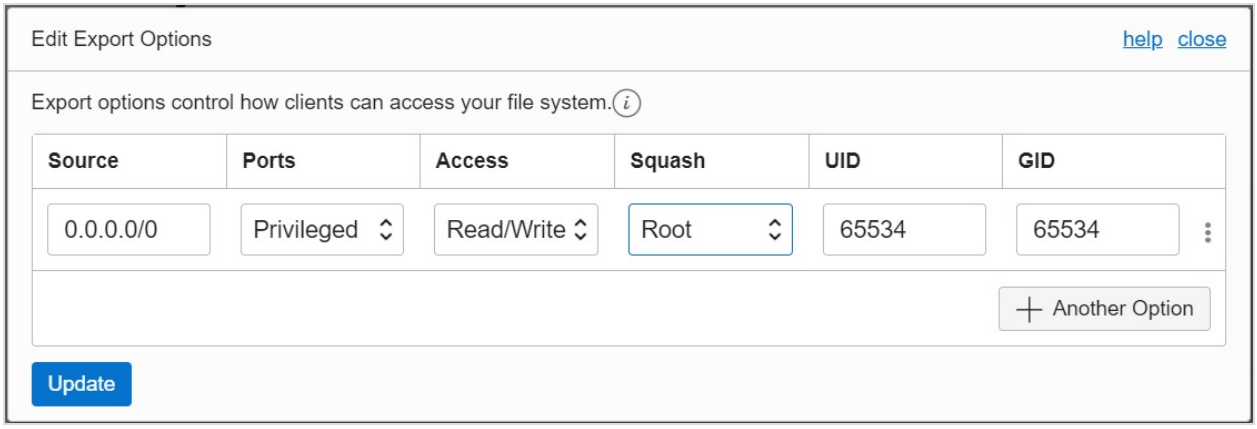
```
PUT/<Current_API_Version>/exports/<File_System_A_export_OCID>
Host: filestorage.us-phoenix-1.oraclecloud.com
<authorization and other headers>
{
    "exportOptions": [
        {
          "source": "10.0.0.8",
          "requirePrivilegedSourcePort": true,
```

```
          "access": "READ_ONLY",
          "identitySquash": "NONE",
          "anonymousUid": 65534,
          "anonymousGid": 65534
        }
    ]
}
```

SCENARIO C: IMPROVE FILE SYSTEM SECURITY

To increase security, you'd like to limit the root user's privileges when connecting to File System A. Use Identity Squash to remap root users to UID/GID 65534. In Unix-like systems, this UID/GID combination is reserved for '*nobody*', a user with no system privileges.

Console Example



CLI Example

```
oci fs export update --export-id <File_System_A_export_OCID> --export-options '
[{"source":"0.0.0.0/0","require-privileged-source-port":"true","access":"READ_
WRITE","identitysquash":"ROOT","anonymousuid":"65534","anonymousgid":"65534"}]'
```

API Example

```
PUT/<Current_API_Version>/exports/<File_System_A_export_OCID>
Host: filestorage.us-phoenix-1.oraclecloud.com
<authorization and other headers>
{
    "exportOptions": [
        {
          "source": "0.0.0.0/0",
          "requirePrivilegedSourcePort": true,
          "access": "READ_WRITE",
          "identitySquash": "ROOT",
          "anonymousUid": 65534,
          "anonymousGid": 65534
        }
    ]
}
```



Tip

If you don't want a file system to be visible to any clients, you can set all of the properties in the `exportOptions` array to empty values. For example,

```
{
   "exportOptions": [
     {
      "source":"",
      "requirePrivilegedSourcePort":"",
      "access": "",
      "identitySquash":""}
]
```

Using the Console

To set export options for a file system

1. Open the navigation menu. Under **Core Infrastructure**, click **File Storage** and then click **File Systems**.
2. In the **List Scope** section, select a compartment. All of the file systems in the selected compartment are displayed.
3. Find the file system you want to set export options for, click the the Actions icon (three dots), and then click **View File System Details**.
4. In the **Exports** list, find the export you want to set export options in, click the the Actions icon (three dots), and then click **View Export Details**. If there is no export listed for the file system, you can create one. See To create an export for a file system for more information.



Tip

To be sure you be sure that you select export, check the following:

- **The export path:** This path uniquely identifies the file system within the mount target. No two exports in a mount target can have the same export path, even if the exports are for the same file system.
- **The mount target name:** File systems can be exported through more than one mount target. Be sure that you've selected the export for the correct mount target.

5. Click **Edit Export Options**.
6. Make one or more of these changes:

- Change an export option entry in the list.
- Click **+Another Option** to create a new export option entry.
- Click the Actions icon (three dots) for an entry and move it up or down in the list.

7. When you're done, click **Update**.

Using the CLI

For information about using the CLI, see Command Line Interface (CLI).

To create an export

Open a command prompt and run `oci fs export create` to create an export for a specified file system within a specified export set. This example creates an export along with its NFS export options.

For example:

```
oci fs export create --export-set-id <export_set_OCID> --file-system-id <file_system_OCID> --path
"</pathname>" --export-options '
[{"source":"10.0.0.0/16","requireprivilegedsourceport":"true","access":"READWRITE","identitysquash":"NO
NE","anonymousuid":"0","anonymousgid":"0"}]'
```



Important

Export Path Names

The path must start with a slash (/) followed by a sequence of zero or more slash-separated elements. For any two export resources associated with the same export set, the path sequence for the first export resource can’t contain the complete path element sequence of the second export sequence. Paths can't end in a slash. No path element can be a period (.) or two periods in sequence (..). Lastly, no path can exceed 255 bytes.

Examples:

Acceptable:

`/example` and `/path`

`/example1` and `/example2`

Not Acceptable:

`/example` and `/example/path`

`/` and `/example`

`/example/`

`/example/path/../example1`

To update export options

Open a command prompt and run `oci fs export update`. To update export options for a specified file system, use `--export-options`.

For example:

```
oci fs export update --export-id <export_OCID> --export-options '[{"source":"<0.0.0.0/0>","require-privileged-source-port":"true","access":"READ_ONLY","identity-squash":"ROOT","anonymous-uid":"65534","anonymous-gid":"65534"}]'

WARNING: Updates to export-options will replace any existing values. Are you sure you want to continue? [y/N]: y
```



Tip

If you don't want a file system to be visible to any clients, you can set all of the properties in Client Options to empty values. For example,

```
oci fs export update --export-id <export_OCID> --export-options '[{"source":"","require-privileged-source-port":"true","access":"READ_ONLY","identity-squash":"ROOT","anonymous-uid":"65534","anonymous-gid":"65534"}]'
```

To list exports

Open a command prompt and run `oci fs export list` to list all exports in a specified compartment.

For example:

```
oci fs export list --compartment-id target_compartment_id
```

To delete an export

Open a command prompt and run `oci fs export delete` to delete an export.

For example:

```
oci fs export delete --export-id export_OCID
```



Warning

When you delete an export, you can no longer mount the file system using the file path specified in the deleted export.

Using the API

- CreateExport
- UpdateExport
- ListExports
- GetExport
- DeleteExport

Creating File Systems

You can create a shared file system in the cloud using the File Storage service. Network access to your file system is provided through a mount target. Exports control how NFS clients access file systems when they connect to a mount target. File systems must have at least one export in one mount target for any instance to mount and use the file system. Typically, you create your first mount target when you create your first file system.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Required IAM Service Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let users create, manage, and delete file systems allows users to create file systems. Since mount targets are network endpoints, users must also have "use" permissions for VNICs, private IPs, private DNS zones, and subnets to create or delete a mount target. See the Policy Reference for more information.

If you're new to policies, see Getting Started with Policies and Common Policies.

Prerequisites

Before you create a file system, you need:

- At least one Virtual Cloud Network (VCN) in a compartment. For more information, see VCNs and Subnets.
- Correctly configured security list rules in the VCN subnet where you plan to create the file system's associated mount target. See Security Lists for information about how security lists work in Oracle Cloud Infrastructure. Use the instructions in Configuring VCN Security List Rules for File Storage to set up security lists for your file systems.

Using the Console

To create a file system

1. Open the navigation menu. Under **Core Infrastructure**, click **File Storage** and then click **File Systems**.

2. In the left-hand navigation, in the **List Scope** section, under **Compartment**, select a compartment.
3. Click **Create File System**.



Note

File systems are encrypted by default. You cannot turn off encryption.

4. You can choose to accept the system defaults, or change them by clicking **Edit Details**.
    - **File System Information:**
        - **Name:**File Storage service creates a default name using "FileSystem-YYMMDD-HHMM". Optionally, change the default name for the file system. It doesn't have to be unique; an Oracle Cloud Identifier (OCID) uniquely identifies the file system.
        - **Availability Domain:** The first availability domain selected in the left panel list is used as default.
    - **Export Information**
      Mount targets use exports to manage access to file systems. The path name uniquely identifies the file system within the mount target, and is used by an instance to mount the file system.
        - **Export Path:** The File Storage service creates a default export path using the file system name. Optionally, replace the default export path name with a new path name, preceded by a forward slash (/). For example, /fss. This value specifies the mount path to the file system (relative to the mount target IP address or hostname). Avoid entering confidential information.



Important

The path must start with a slash (/) followed by a sequence of zero or more slash-separated elements. For multiple file systems associated with a single mount target, the path sequence for the first file system cannot contain the complete path element sequence of the second file system path sequence. Paths cannot end in a slash. No path element can be a period (.) or two periods in sequence (..). Lastly, no path can exceed 255 bytes. For example:

Acceptable:

`/example` and `/path`

`/example` and `/example2`

Not Acceptable:

`/example` and `/example/path`

`/` and `/example`

`/example/`

`/example/path/../example1`



Warning

If one file system associated to a mount target has '/' specified as an export path, you can't associate another file system with that mount target.



Note

Export paths cannot be edited after the export is created. If you want to use a different export path, you must create a new export with the desired path. Optionally, you can then delete the export with the old path.

For more information, see Paths in File Systems.

- **Use Secure Export Options:** Select to set the export options to require NFS clients to use a privileged port (1-1023) as its source port. This option enhances security because only a client with root privileges can use a privileged source port. After the export is created, you can edit the export options to adjust security. See Working with NFS Export Options for more information.



Warning

Leaving the "Use Secure Export Options" setting disabled allows unprivileged users to read and modify any file or directory on the target file system.

- **Mount Target Information:**
  File systems must be associated with a mount target to be mounted by an instance.
  If you have one or more previously created mount targets in the availability domain, the File Storage service automatically chooses the most recently created mount target in the list. If you don't have a mount target in the selected availability domain, the File Storage service creates one using the following defaults.
  - **Mount Target Name:** File Storage service creates a default mount target name using "Mount-YYYYMMDD-HHMM".
  - **Compartment:** The compartment you're currently working in.
  - **Virtual Cloud Network:** The first VCN listed in the current compartment is used as default.
  - **Subnet:** The most recently created subnet listed in the selected availability domain is used as default. Subnets can be either AD-specific or regional (regional ones have "*regional*" after the name). For more information, see About Regional Subnets.

5. If you want to *accept the defaults* for the mount target, click **Create**. *The file system is created with the information displayed*. If you want to choose another mount target or change the default information, click the **Edit Details** link.
6. In the **Mount Target Information** section, specify details for the mount target that is associated with the file system:

- **Select an Existing Mount Target:** Choose this option if you want to associate the file system with a mount target you already created. Choose the **Mount Target** from the list. Click the **click here** link in the dialog box if you want to enable compartment selection for the mount target.

  Tip

  If there aren't any mount targets in the current combination of availability domain and compartment, this option is disabled. You can:

  - Choose a different compartment.
  - Choose a different availability domain in the *File System Information* section.
  - Create a new mount target.

- **Create a New Mount Target:** Choose this option if you want to create a new mount target associated with this file system. By default, the mount target is created in your current compartment and you can use network resources in that compartment. Click the **click here** link in the dialog box if you want to enable compartment selection for the mount target, its VCN, or subnet resources.



Important

The mount target is always in the same availability domain as the file system. While it is possible to access mount targets from any AD in a region, for optimal performance, your mount target and file system should be in the same availability domain as the Compute instances that access them. For more information, see Regions and Availability Domains.

- **Create in Compartment:** Specify the compartment you want to create the mount target in.
- **New Mount Target Name:** Optionally, replace the default with a friendly name for the mount target. It doesn't have to be unique; an Oracle Cloud Identifier (OCID) uniquely identifies the mount target. Avoid entering confidential information.



Note

The mount target name is different than the DNS hostname, which is specified in step 7.

- **Virtual Cloud Network Compartment:** The compartment containing the cloud network (VCN) in which to create the mount target.
- **Virtual Cloud Network:** Select the cloud network (VCN) where you want to create the new mount target.

- **Subnet Compartment:** Specify the compartment containing a subnet within the VCN to attach the mount target to.
- **Subnet:** Select a subnet to attach the mount target to. Subnets can be either AD-specific or regional (regional ones have "*regional*" after the name). For more information, see About Regional Subnets.



Warning

Each mount target requires three internal IP addresses in the subnet to function. Do not use /30 or smaller subnets for mount target creation because they do not have sufficient available IP addresses. Two of the IP addresses are used during mount target creation. The third IP address must remain available for the mount target to use for high availability failover.

7. Optionally, click **Show Advanced Options** to configure the mount target's advanced options.
   - **IP Address:** You can specify an unused IP address in the subnet you selected for the mount target.
   - **Hostname:** You can specify a hostname you want to assign to the mount target.



Note

The File Storage service constructs a fully qualified domain name (FQDN) by combining the hostname with the FQDN of the subnet the mount target is located in.

For example, `myhostname.subnet123.dnslabel.oraclevcn.com`.

Once created, the hostname may be changed in the mount target's Details page. See Managing Mount Targets for more information.

8. Click **Create**.

The File Storage service typically creates the file system and mount target within seconds. Next, mount the file system from an instance so that you can read and write directories and files in your file system. See Mounting File Systems for instructions about obtaining mount commands for your operating system type and mounting your file system.

Using the command line interface (CLI)

For information about using the CLI, see Command Line Interface (CLI).

To create a file system

Open a command prompt and run `oci fs file-system create` to create a file system. For example:

```
oci fs file-system create --availability-domain <target_availability_domain> --display-name "<My File System>" --compartment-id <target_compartment_id>
```



Warning

Avoid entering confidential information in the file system `display-name`.

The file system is created.

To create a mount target

You can create a mount target for file systems in a specified compartment and subnet. A file system can only be associated with a mount target in the same availability domain.



Warning

Each mount target requires three internal IP addresses in the subnet to function. Do not use /30 or smaller subnets for mount target creation because they do not have sufficient available IP addresses. Two of the IP addresses are used during mount target creation. The third IP address must remain available for the mount target to use for high availability failover.

Open a command prompt and run `oci fs mount-target create` to create a mount target.

For example:

```
oci fs mount-target create --availability-domain <target_availability domain> --compartment-id <target_
compartment_id> --subnet-id <subnet_OCID> --display-name "<My Mount Target>"
```



Warning

Avoid entering confidential information in the mount target `display-name`.

To create an export

An export is a file system together with the path that can be used to mount it. Each export resource belongs to one export set.

Open a command prompt and run `oci fs export create` to create an export for a specified file system within a specified export set.

For example:

```
oci fs export create --export-set-id <export_set_OCID> --file-system-id <file_system_OCID> --path
"</pathname>"
```

Important

The path must start with a slash (/) followed by a sequence of zero or more slash-separated elements. For multiple file systems associated with a single mount target, the path sequence for the first file system cannot contain the complete path element sequence of the second file system path sequence. Paths cannot end in a slash. No path element can be a period (.) or two periods in sequence (..). Lastly, no path can exceed 255 bytes. For example:

Acceptable:

`/example` and `/path`

`/example` and `/example2`

Not Acceptable:

`/example` and `/example/path`

`/` and `/example`

`/example/`

`/example/path/../example1`

Warning

If one file system associated to a mount target has '/' specified as an export path, you can't associate another file system with that mount target.



Note

Export paths cannot be edited after the export is created. If you want to use a different export path, you must create a new export with the desired path. Optionally, you can then delete the export with the old path.

For more information, see Paths in File Systems.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operations to create file systems:

- CreateFileSystem
- CreateMountTarget
- CreateExport

Mounting File Systems

Users of Unix-style operating systems and Windows Server 2008 R2, 2012 R2, or 2016 can connect to a file system and write files. Mount targets serve as file system network access points for file systems. After your mount target is assigned an IP address, you can use it together with the file system export path to mount the file system. On the instance from which you want to mount the file system, you need to install an NFS client. For Unix-style operating systems, you create a mount point. When you mount the file system, the mount point effectively represents the root directory of the File Storage file system, allowing you to write

files to the file system from the instance. Windows operating systems use a drive letter assignment instead of a mount point to represent root access.

Prerequisites

- The file system must have at least one export in at least one mount target. When you create a new file system, an export for the file system is created at the same time. See Creating File Systems for more information.
- Correctly configured security list rules in the VCN subnet where the file system's associated mount target resides. See Security Lists for information about how security lists work in Oracle Cloud Infrastructure. Use the instructions in Configuring VCN Security List Rules for File Storage to set up security lists for your file systems.

Mounting File Systems From an Instance

Mounting File Systems From Unix-Style Instances (Including Oracle Linux DB instances)

Mounting File Systems From Windows Instances

Obtaining Mount Command Samples

You can use the Console to get mount command samples that include all the information for a specific mount target and file system. Samples are available for the following operating system images:

- Oracle Linux
- CentOS
- Debian
- Red Hat Linux
- Ubuntu



Warning

When mounting file systems, the following mount option combination is **not supported** by the File Storage service:

- `soft` when the file system is mounted with the read/write mount option (`-o rw`). **This combination can cause corruption of your data**.

The following mount options or mount option combinations are **not recommended** for use with the File Storage service:

- `soft` when the file system is mounted with the read-only mount option (`-o ro`) and the `timeo` has been specified as less than `300` seconds. **This combination can cause a profusion of I/O error responses.**
- `nolock`, `rsize`, or `wsize`. **These options cause issues with performance and file locking.**

Required IAM Service Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let users create, manage, and delete file systems allows users to obtain mount commands.

If you're new to policies, see Getting Started with Policies and Common Policies.

Using the Console

To get mount command samples

1. Open the navigation menu. Under **Core Infrastructure**, click **File Storage** and then click **File Systems**.
2. In the **List Scope** section, select a compartment.
   The Console displays a list of file systems that have already been created in the compartment, if any.
3. Find the file system you want to mount, click the Actions icon (three dots), and then click **View File System Details**.
4. In **Resources**, click **Exports**.
5. Find the export in the mount target you want to use to mount the file system, click the Actions icon (three dots), and then click **Mount Commands**.



Tip

To be sure that you select the correct export, check the following:

- **The export path:** This path uniquely identifies the file system within the mount target. No two exports in a mount target can have the same export path, even if the exports are for the same file system.
- **The mount target name:** File systems can be exported through more than one mount target. Be sure that you've selected the export for the correct mount target.

6. In **Image**, choose the image of the Compute instance you want to mount the file system to.
7. Click the **Copy** link to copy the commands.

Next, mount the file system from a Unix-style or Windows instance.

Mounting File Systems From Unix-Style Instances

Users of Ubuntu and Linux operating systems can use the command line to connect to a file system and write files. Mount targets serve as network access points for file systems. After your mount target is assigned an IP address, you can use it together with the export path to mount the file system. On the instance from which you want to mount the file system, you need to install an NFS client and create a mount point. When you mount the file system, the mount point effectively represents the root directory of the File Storage file system, allowing you to write files to the file system from the instance.

Prerequisites

- The file system must have at least one export in at least one mount target. When you create a new file system, an export for the file system is created at the same time. See Creating File Systems for more information.
- Correctly configured security list rules in the VCN subnet where the file system's associated mount target resides. See Security Lists for information about how security lists work in Oracle Cloud Infrastructure. Use the instructions in Configuring VCN Security List Rules for File Storage to set up security lists for your file systems.

Mounting File Systems

You can use the following instructions to construct your mount commands, or use the Console to get mount command samples that include all the information for a specific mount target and file system. For more information, see To get mount command samples.



Warning

When mounting file systems, the following mount option combination is **not supported** by the File Storage service:

- `soft` when the file system is mounted with the read/write mount option (`-o rw`). **This combination can cause corruption of your data**.

The following mount options or mount option combinations are **not recommended** for use with the File Storage service:

- `soft` when the file system is mounted with the read-only mount option (`-o ro`) and the `timeo` has been specified as less than `300` seconds. **This combination can cause a profusion of I/O error responses.**
- `nolock`, `rsize`, or `wsize`. **These options cause issues with performance and file locking.**

To mount a file system from Ubuntu or Debian

1. Open a command window. Then, get the NFS client by copying and pasting the **Install Command** from the Console or type the following:

   ```
   sudo apt-get install nfs-common
   ```

2. Create a mount point by copying and pasting the **Create Mount Point Command** from the Console or type the following, replacing `yourmountpoint` with the local directory from which you want to access your file system.

```
sudo mkdir -p /mnt/yourmountpoint
```

3. Mount the file system by copying and pasting the **Mount Command** from the Console or type the following. Replace `10.x.x.x:` with the local subnet IP address assigned to your mount target, `fs-export-path` with the export path you specified when associating the file system with the mount target, and `yourmountpoint` with the path to the local mount point. The export path is the path to the file system (relative to the mount target IP address or hostname). If you did not specify a path when you associated the file system and mount target, then `10.x.x.x:/` represents the full extent of the mount target.



Tip

IP address and export path information is available in the **Details** page of the mount target associated with your file system. See To view details of a mount target for more information.

```
sudo mount -o nosuid,resvport 10.x.x.x:/fs-export-path /mnt/yourmountpoint
```



Warning

Omitting the `-o nosuid` option may allow unprivileged users to escalate their permissions to 'root'. The `nosuid` option disables set-user-identifier or set-group-identifier bits within the mounted system, which are rarely used.



Note

The `-o resvport` option is required when the "Require Privileged Source Port" export option is used and otherwise optional. It causes the mounting filesystem to connect from a privileged source port (1-1023). See Working with NFS Export Options for more information.

4. View the file system.

```
df -h
```

5. Write a file to the file system by typing the following. Replace `yourmountpoint` with the path to the local mount point and `helloworld` with your file name.

```
sudo touch /mnt/yourmountpoint/helloworld
```

6. Verify that you can view the file by typing the following. Replace `yourmountpoint` with the path to the local mount point.

```
cd /mnt/yourmountpoint
```

```
ls
```

See Mount Command Fails in Troubleshooting Your File System for more information about common issues you may encounter.

To mount a file system from Linux, Red Hat, or CentOS

1. Open a command window. Then, get the NFS client by copying and pasting the **Install Command** from the Console or typing the following:

```
sudo yum install nfs-utils
```

2. Create a mount point by copying and pasting the **Create Mount Point Command** from the Console or type the following, replacing `yourmountpoint` with the local directory

from which you want to access your file system.

```
sudo mkdir -p /mnt/yourmountpoint
```

3. Mount the file system by copying and pasting the **Mount Command** from the Console or type the following. Replace `10.x.x.x:` with the local subnet IP address assigned to your mount target, `fs-export-path` with the export path you specified when associating the file system with the mount target, and `yourmountpoint` with the path to the local mount point. The export path is the path to the file system (relative to the mount target's IP address or hostname). If you did not specify a path when you associated the file system and mount target, then `10.x.x.x:/` represents the full extent of the mount target.



Tip

IP address and export path information is available in the **Details** page of the mount target associated with your file system. See To view details of a mount target for more information.

```
sudo mount -o nosuid,resvport 10.x.x.x:/fs-export-path /mnt/yourmountpoint
```



Warning

Omitting the `-o nosuid` option may allow unprivileged users to escalate their permissions to 'root'. The `nosuid` option disables set-user-identifier or set-group-identifier bits within the mounted system, which are rarely used.



Note

The `-o resvport` option is required when the "Require Privileged Source Port" export option is used and otherwise optional. It causes the mounting filesystem to connect from a privileged source port (1-1023). See Working with NFS Export Options for more information.

4. View the file system.

```
df -h
```

5. Write a file to the file system by typing the following. Replace `yourmountpoint` with the path to the local mount point and `helloworld` with your file name.

```
sudo touch /mnt/yourmountpoint/helloworld
```

6. Verify that you can view the file by typing the following. Replace `yourmountpoint` with the path to the local mount point.

```
cd /mnt/yourmountpoint
```

```
ls
```

See Mount Command Fails in Troubleshooting Your File System for more information about common issues you may encounter.

To mount a file system from a Database VM instance

Database VM instances are built on Oracle Linux 6.8, unlike Oracle Linux Compute instances, which run on version 7.4. The NFS Utilities package is pre-installed on DB instances, but the Open Network Computing Remote Procedure Call (ONC RPC) rpcbind utility is disabled by default. Oracle Linux 6.8 does not have systemd, so DB instances are managed differently than OL compute instances. An Oracle DB instance comes with a set of iptables rules that excludes any non-database ports and need to be updated to allow mount target traffic.

1. SSH to the DB system.

```
ssh -i <private_key_path> opc@<db_system_ip_address>
```

2. Start the rpcbind service by typing the following:

```
sudo service rpcbind start
```

3. Use the chkconfig command to enable starting rpcbind service at system startup.

```
sudo chkconfig rpcbind on
```

4. Change the default configuration of iptables to include the mount target IP address and allow traffic by typing the following. Replace `10.x.x.x` with the local subnet address assigned to the mount target for the file system. Save the new iptables entries.

```
sudo iptables -A INPUT -p tcp -s 10.x.x.x -j ACCEPT
```

```
sudo iptables -A OUTPUT -p tcp -s 10.x.x.x -j ACCEPT
```

```
sudo service iptables save
```

5. Create a mount point by typing the following, replacing `yourmountpoint` with the local directory from which you want to access your file system.

```
sudo mkdir -p /mnt/yourmountpoint
```

6. Mount the file system by typing the following. Replace `10.x.x.x:` with the local subnet IP address assigned to your mount target, `fs-export-path` with the export path you specified when associating the file system with the mount target, and `yourmountpoint` with an absolute path to a local mount point. The export path is the path to the file system (relative to the mount target IP address or hostname). If you did not specify a path when you associated the file system and mount target, then `10.x.x.x:/` represents the full extent of the mount target.



Tip

IP address and export path information is available in the **Details** page of the mount target associated with your file system. See To view details of a mount target for more information.

```
sudo mount -t nfs -o nosuid,resvport,tcp,vers=3 10.x.x.x:/fs-export-path /mnt/yourmountpoint
```



Warning

Omitting the `-o nosuid` option may allow unprivileged users to escalate their permissions to 'root'. The `nosuid` option disables set-user-identifier or set-group-identifier bits within the mounted system, which are rarely used.



Note

The `-o resvport` option is required when the "Require Privileged Source Port" export option is used and otherwise optional. It causes the mounting filesystem to connect from a privileged source port (1-1023). See Working with NFS Export Options for more information.

See Mount Command Fails in Troubleshooting Your File System for more information about common issues you may encounter.

To auto-mount a shared file system

Auto-mount ensures that a file system is automatically re-mounted on an instance if it is rebooted.

1. Open a command window. Then, mount the file system using the steps described in the previous section.
2. Type the following command to get the file system entry point:

   ```
   sudo cat /etc/mtab |grep -i nfs
   ```

3. Copy the file system entry point, and open the /etc/fstab file:

   ```
   cd /etc
   ```

   ```
   vi fstab
   ```

4. Add the following line to the fstab file:

   ```
   <file_system_ip_address>:<file_system_path_name><your_local_mount_point> nfs
   defaults,nofail,nosuid,resvport 0 0
   ```



Warning

Omitting the `-o nosuid` option may allow unprivileged users to escalate their permissions to 'root'. The `nosuid` option disables set-user-identifier or set-group-identifier bits within the mounted system, which are rarely used.



Important

Be sure to add the `nofail` option to each entry. This option ensures that an unavailable file system does not cause the instance reboot process to fail.



Note

The `-o resvport` option is required when the "Require Privileged Source Port" export option is used and otherwise optional. It causes the mounting filesystem to connect from a privileged source port (1-1023). See Working with NFS Export Options for more information.

5. Save the fstab file.

See Mount Command Fails in Troubleshooting Your File System for more information about common issues you may encounter.

Mounting File Systems From Windows Instances

Users of Windows Server 2008 R2, 2012 R2, or 2016 can mount a file system on any available drive letter using the mount target IP address and the file system export path.

On the instance from which you want to mount the file system, you need to install the Windows NFS client.



Warning

Installing the Windows NFS client may require a restart of your system.

Access to NFS file systems requires UNIX-style user and group identities, which are not the same as Windows user and group identities. To enable users to access NFS shared resources, Windows client for NFS accesses file systems anonymously, using 'AnonymousGid' and 'AnonymousUid'. On brand new file systems, write permissions are only granted to the root user. The 'AnonymousGid' and 'AnonymousUid' identity values must be changed to allow write access.



Warning

Updating the 'AnonymousGid' and 'AnonymousUid' values require registry changes to your system.

After you have installed the NFS client and correctly mapped user identities, you can mount the file system to any available drive letter using the command line or **Map network drive**. You can access your file system through the chosen drive letter to write files.

Prerequisites

- The file system must have at least one export in at least one mount target. When you create a new file system, an export for the file system is created at the same time. See Creating File Systems for more information.
- Correctly configured security list rules in the VCN subnet where the file system's associated mount target resides. See Security Lists for information about how security lists work in Oracle Cloud Infrastructure. Use the instructions in Configuring VCN Security List Rules for File Storage to set up security lists for your file systems.



Warning

When mounting file systems, the following mount option combination is **not supported** by the File Storage service:

- `soft` when the file system is mounted with the read/write mount option (`-o rw`). **This combination can cause corruption of your data**.

The following mount options or mount option combinations are **not recommended** for use with the File Storage service:

- `soft` when the file system is mounted with the read-only mount option (`-o ro`) and the `timeo` has been specified as less than `300` seconds. **This combination can cause a profusion of I/O error responses.**
- `nolock`, `rsize`, or `wsize`. **These options cause issues with performance and file locking.**

Using Windows Command Prompt

To mount a file system from Windows Server 2008 R2 Command Prompt

1. Install Services for NFS components.
    a. Click **Start** go to **Administrative Tools**, then click **Server Manager**.
    b. In the left pane, click **Roles**.

c. Under **Roles Summary** in the right pane, click **Add Roles**. The **Add Roles Wizard** appears. Click **Next**.

d. Select the **File Services** check box from the list and click **Next**. Review the information and click **Next**.

e. Select the **Services for Network File System** check box from the list, and click **Next**.

f. Confirm your selections, then click **Install**. Click **Close** when the installation is complete.

2. Open the registry editor (regedit) and map the AnonymousGid and AnonymousUid to the root user.



Warning

User identity mapping requires changes to your system registry.

a. Click **Windows Search.**

b. Enter `regedit` in the **Search** field and press **Enter**.

c. Click **Yes** to allow changes to your device.

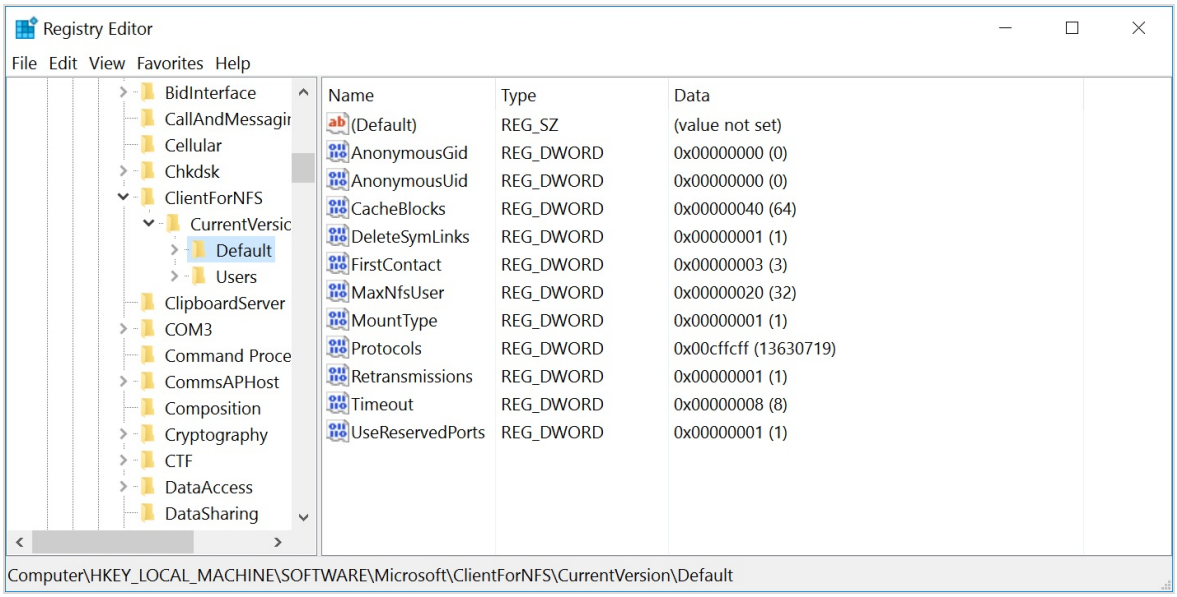
d. Click `HKEY_LOCAL_MACHINE`. Then, browse to: `Software\Microsoft\ClientForNFS\CurrentVersion\Default`.

3. Add a new DWORD32 registry entry for `AnonymousGid`:

a. Click **Edit**, and select **New DWORD (32 bit) Value**.

b. In the **Name** field, enter `AnonymousGid`. Leave the value at `0`.

4. Repeat step 3 to add a second DWORD32 registry entry named `AnonymousUid` with a value of `0`.



5. Open Windows Command Line (CMD) and run as **Administrator**:
    a. Click **Start** .
    b. Press **Ctrl+Shift** and click **Command Prompt**.
    c. Click **Yes**.

> **Important**
>
> If you've set export options for your file system to require clients to connect from a privileged source port (1-1023), then you must set the **UseReserverdPorts** registry key to **1**.
> For more information, see Working with NFS Export Options.

6. In the Administrator: Windows Command Prompt (CMD) window, restart the NFS Client by typing the following:

```
nfsadmin client stop
```

```
nfsadmin client start
```

7. Close the Administrator: Windows Command Prompt (CMD) window. Open a **standard** Command Prompt Window:
   a. Click **Start**, then click **Command Prompt**.



Important

NFS file systems mounted as Administrator are not available to standard users.

8. In the standard Windows Command Line (CMD) window, mount the file system by typing the following. Replace `10.x.x.x:` with the local subnet IP address assigned to your mount target, `fs-export-path` with the export path you specified when associating the file system with the mount target, and `X` with the drive letter of any available drive you want to map the file system to.



Tip

IP address and export path information is available in the **Details** page of the mount target associated with your file system. See To view details of a mount target for more information.

```
mount 10.x.x.x:/fs-export-path X:
```



Important

The export path is the path to the file system (relative to the mount target IP address or hostname). If you did not specify a path when you associated the file system and mount target, then "/" represents the full extent of the mount target. In that case, you must use a "!" when mounting the file system. For example: `mount 10.0.0.0:/! X:`

9. Write a file to the file system by typing the following. Replace `X` with the drive letter you used in step 8 and `helloworld` with your file name.

```
X:
```

```
echo > helloworld.txt
```

10. Verify that you can view the file by typing the following.

```
dir
```

See Troubleshooting Windows NFS Client Connections for more information on common issues you may encounter.

To mount a file system from Windows Server 2012 R2 or 2016 Command Prompt

1. Open **Windows PowerShell** and run as **Administrator**:
    a. Go to **Start** and click the **Windows PowerShell** icon.
    b. In Windows PowerShell, type the following to run as Administrator:

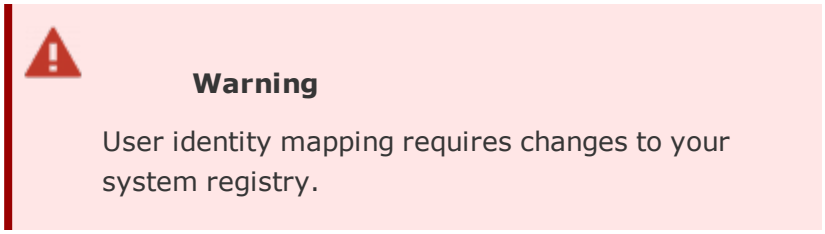
```
Start-Process powershell -Verb runAs
```

c. In the **User Account Control** window, click **Yes**. A new Administrator: PowerShell window opens. You can close the standard PowerShell window to avoid confusing them.

2. In Administrator: PowerShell, get the NFS client by typing the following:

```
Install-WindowsFeature -Name NFS-Client
```

3. If necessary, restart your system.
4. Open the registry editor (regedit) to map the AnonymousGid and AnonymousUid to the root user.



Warning

User identity mapping requires changes to your system registry.

a. Click **Windows Search.**

b. Enter `regedit` in the **Search** field and press **Enter**.

c. Click **Yes** to allow changes to your device.

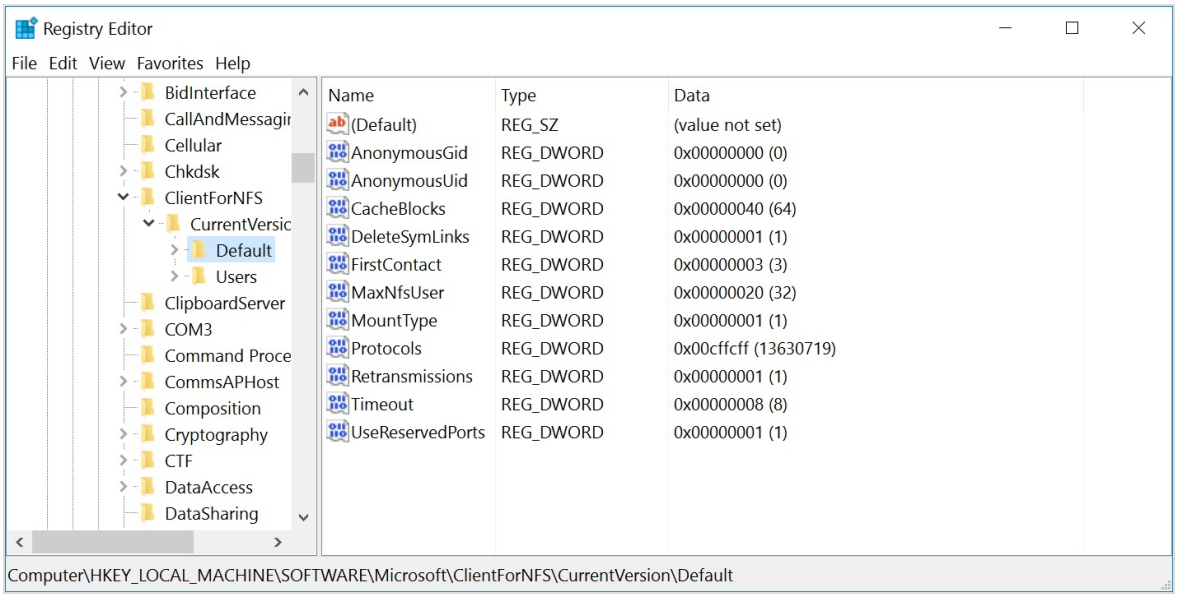
d. Click `HKEY_LOCAL_MACHINE`. Then, browse to: `Software\Microsoft\ClientForNFS\CurrentVersion\Default`.

5. Add a new DWORD32 registry entry for `AnonymousGid`:

a. Click **Edit**, and select **New DWORD (32 bit) Value**.

b. In the **Name** field, enter `AnonymousGid`. Leave the value at `0`.

6. Repeat step 5 to add a second DWORD32 registry entry named `AnonymousUid` with a value of `0`.



Important

If you've set export options for your file system to require clients to connect from a privileged source port (1-1023), then you must set the **UseReserverdPorts** registry key to **1**.

For more information, see Working with NFS Export Options.

7. Open Windows Command Line (CMD) and run as Administrator:



Tip

To avoid confusion, be sure to close the Windows PowerShell application before opening the Windows Command Line application. The following commands don't work in Windows PowerShell.

a. Go to **Start** and scroll down to **Apps**.

b. In the **Windows System** section, press **Ctrl+Shift** and click **Command Prompt**.

8. In the Windows Command Line (CMD) window, restart the NFS Client by typing the following:

```
nfsadmin client stop
```

```
nfsadmin client start
```

9. Close the Administrator: Windows Command Prompt (CMD) window. Open a **standard** Command Prompt Window:

a. Click **Start**, then click **Command Prompt**.



Important

NFS file systems mounted as Administrator are not available to standard users.

10. In the standard Windows Command Line (CMD) window, mount the file system by typing the following. Replace `10.x.x.x:` with the local subnet IP address assigned to your mount target, `fs-export-path` with the export path you specified when associating the file system with the mount target, and `X` with the drive letter of any available drive you

want to map the file system to.



Tip

IP address and export path information is available in the **Details** page of the mount target associated with your file system. See To view details of a mount target for more information.

```
mount 10.x.x.x:/fs-export-path X:
```



Important

The export path is the path to the file system (relative to the mount target IP address or hostname). If you did not specify a path when you associated the file system and mount target, then "/" represents the full extent of the mount target. In that case, you must use a "!" when mounting the file system. For example: `mount 10.0.0.0:/! X:`

11. Write a file to the file system by typing the following. Replace `X` with the drive letter you used in step 10 and `helloworld` with your file name.

```
X:
```

```
echo > helloworld.txt
```

12. Verify that you can view the file by typing the following.

```
dir
```

See Troubleshooting Windows NFS Client Connections for more information on common issues you may encounter.

Using Windows File Explorer

To mount a file system from Windows Server 2008 R2 File Explorer

1. Install Services for NFS components.
    a. Click **Start** go to **Administrative Tools**, then click **Server Manager**.
    b. In the left pane, click **Roles**.
    c. Under **Roles Summary** in the right pane, click **Add Roles**. The **Add Roles Wizard** appears. Click **Next**.
    d. Select the **File Services** check box from the list and click **Next**. Review the information and click **Next**.
    e. Select the **Services for Network File System** check box from the list, and click **Next**.
    f. Confirm your selections, then click **Install**. Click **Close** when the installation is complete.
2. Open the registry editor (regedit) to map the AnonymousGid and AnonymousUid to the root user.



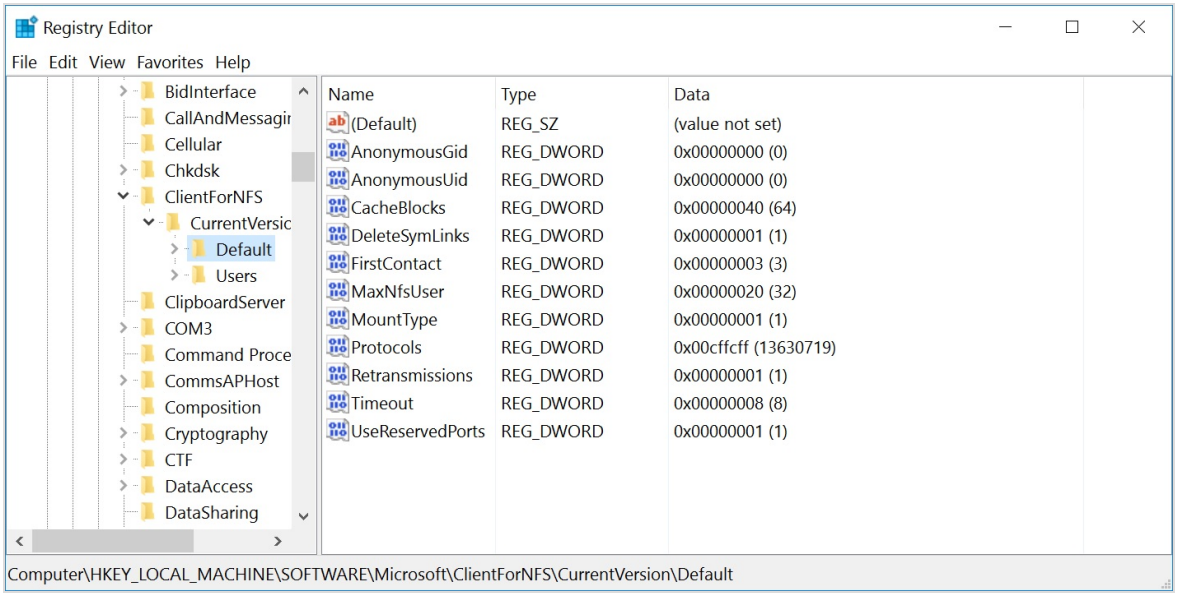
Warning

User identity mapping requires changes to your system registry.

    a. Click **Windows Search.**
    b. Enter `regedit` in the **Search** field and press **Enter**.
    c. Click **Yes** to allow changes to your device.

d. Click `HKEY_LOCAL_MACHINE`. Then, browse to: `Software\Microsoft\ClientForNFS\CurrentVersion\Default`.

3. Add a new DWORD32 registry entry for `AnonymousGid`:
   a. Click **Edit**, and select **New DWORD (32 bit) Value**.
   b. In the **Name** field, enter `AnonymousGid`. Leave the value at `0`.
4. Repeat step 3 to add a second DWORD32 registry entry named `AnonymousUid` with a value of `0`.



> **Important**
>
> If you've set export options for your file system to require clients to connect from a privileged source port (1-1023), then you must set the **UseReserverdPorts** registry key to **1**.
> For more information, see Working with NFS Export Options.

5. Open Windows Command Line (CMD) and run as **Administrator**:
   a. Click **Start** .
   b. Press **Ctrl+Shift** and click **Command Prompt**.
   c. Click **Yes**.
6. In the Administrator: Windows Command Prompt (CMD) window, restart the NFS Client by typing the following:

```
nfsadmin client stop
```

```
nfsadmin client start
```

7. Open **File Explorer** and select **Computer**. Click the **Map network drive** tab.
8. Select the **Drive** letter that you want to assign to the file system.
9. In the **Folder** field, enter the following. Replace `10.x.x.x` with the local subnet IP address assigned to your mount target, and `fs-export-path` with the export path you specified when associating the file system with the mount target.



Tip

IP address and export path information is available in the **Details** page of the mount target associated with your file system. See To view details of a mount target for more information.

```
\\10.x.x.x\fs-export-path
```



Important

The export path is the path to the file system (relative to the mount target IP address or hostname). If you did not specify a path when you associated the file system and mount target, then "\" represents the full extent of the mount target. In that case, you must use a "!" when entering the file system folder path. For example: `\\10.0.0.0\!`

10. Click the **Finish** button when complete.

See Troubleshooting Windows NFS Client Connections for more information on common issues you may encounter.

To mount a file system from Windows Server 2012 R2 or 2016 File Explorer

1. Open **Windows PowerShell** and run as **Administrator**:
    a. Go to **Start** and click the **Windows PowerShell** icon.
    b. In Windows PowerShell, type the following to run as Administrator:

```
Start-Process powershell -Verb runAs
```

c. In the **User Account Control** window, click **Yes**. A new Administrator: PowerShell window opens. You can close the standard PowerShell window to avoid confusing them.

2. In Administrator: PowerShell, get the NFS client by typing the following:

```
Install-WindowsFeature -Name NFS-Client
```

3. If necessary, restart your system.
4. Open the registry editor (regedit) to map the AnonymousGid and AnonymousUid to the root user.

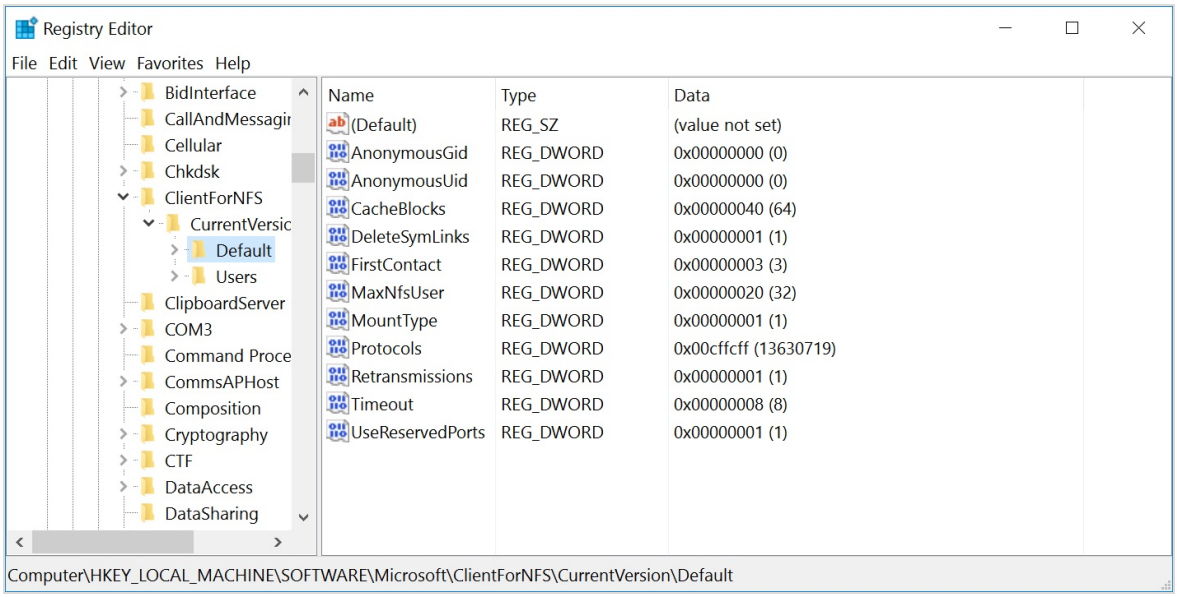


Warning

User identity mapping requires changes to your system registry.

a. Click **Windows Search.**
b. Enter `regedit` in the **Search** field and press **Enter**.
c. Click **Yes** to allow changes to your device.
d. Click `HKEY_LOCAL_MACHINE`. Then, browse to: `Software\Microsoft\ClientForNFS\CurrentVersion\Default`.

5. Add a new DWORD32 registry entry for `AnonymousGid`:
   a. Click **Edit**, and select **New DWORD (32 bit) Value**.
   b. In the **Name** field, enter `AnonymousGid`. Leave the value at `0`.
6. Repeat step 5 to add a second DWORD32 registry entry named `AnonymousUid` with a value of `0`.



Important

If you've set export options for your file system to require clients to connect from a privileged source port (1-1023), then you must set the **UseReserverdPorts** registry key to **1**.

For more information, see Working with NFS Export Options.

7. Open Windows Command Line (CMD) and run as Administrator:
   a. Go to **Start** and scroll down to **Apps**.
   b. In the **Windows System** section, press **Ctrl+Shift** and click **Command Prompt**.
8. In the Windows Command Line (CMD) window, restart the NFS Client by typing the

following:

```
nfsadmin client stop
```

```
nfsadmin client start
```

9. Open **File Explorer** and select **This PC**. In the **Computer** tab, select **Map network drive**.
10. Select the **Drive** letter that you want to assign to the file system.
11. In the **Folder** field, enter the following. Replace `10.x.x.x` with the local subnet IP address assigned to your mount target, and `fs-export-path` with the export path you specified when associating the file system with the mount target.



Tip

IP address and export path information is available in the **Details** page of the mount target associated with your file system. See To view details of a mount target for more information.

```
\\10.x.x.x\fs-export-path
```



Important

The export path is the path to the file system (relative to the mount target IP address or hostname). If you did not specify a path when you associated the file system and mount target, then "\" represents the full extent of the mount target. In that case, you must use a "!" when entering the file system folder path. For example: `\\10.0.0.0\!`

12. Click the **Finish** button when complete.

See Troubleshooting Windows NFS Client Connections for more information on common issues you may encounter.

Managing File Systems

In the File Storage service, file systems are associated with a single compartment. When you select a compartment, the Console displays all file systems in the compartment. You can also see exports and snapshots associated with each file system. If there are no file systems in the compartment, see Creating File Systems for instructions about creating one.

The compartment has policies that indicate what actions a user can take to manage file system. UNIX permissions control what actions a user can take on the files stored in the file system. See About Security for more information.

Actions you can take to manage a file system include:

- Viewing file system details
- Editing file system settings
- Viewing associated file system resources
- Creating an export for the file system
- Deleting a file system

You can perform most administrative tasks for your file systems using the Console, Command Line Interface (CLI), or API. You can use the Console to list mount targets exporting a specific file system. Use the API or CLI if you want to list all mount targets in a compartment.

To access a file system, it must have at least one export in one mount target. Next, mount the file system from an instance, and then you can create directories and read and write files. For more information about creating an export for a file system, see To create an export for a file system in this topic. For more information about accessing your file system, see Mounting File Systems.

Required IAM Service Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let users create, manage, and delete file systems allows users to manage file systems.

If you're new to policies, see Getting Started with Policies and Common Policies.

Tagging Resources

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

Details About Your File System

The file system details page provides the following information about your file system:

File System OCID

Every Oracle Cloud Infrastructure resource has an Oracle-assigned unique ID called an Oracle Cloud Identifier (OCID). You need your file system's OCID to use the Command Line Interface (CLI) or the API. You also need the OCID when contacting support.

Availability Domain

When you create a file system, you specify the availability domain that it resides in. An availability domain is one or more data centers located within a region. You need your file system's availability domain to use the Command Line Interface (CLI) or the API. For more information, see Regions and Availability Domains.

CREATED

The date and time that the file system was created.

COMPARTMENT

When you create a file system, you specify the compartment that it resides in. A compartment is a collection of related resources (such as cloud networks, compute instances, or file systems) that are only accessible to those groups that have been given permission by an administrator in your organization. You need your file system's compartment to use the Command Line Interface (CLI) or the API. For more information, see Managing Compartments.

UTILIZATION

Metered size of the file system that gets updated hourly.

> **Important**
>
> **There can be a delay of up to 1 hour** when reporting file system usage.

RESOURCES

Resources such as exports and snapshots that are associated with the file system are listed here. Click the resource type link to see a list of each individual resource. Each export in the list shows the file system's export path and mount target. You need the export path to mount a file system.

> **Warning**
>
> Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Using the Console

To view file system details

The File Storage service displays a list of file systems in each compartment.

1. Open the navigation menu. Under **Core Infrastructure**, click **File Storage** and then click **File Systems**.
2. In the **List Scope** section, select a compartment.
3. To view information about a file system, find the file system, click the Actions icon (three dots), and then click **View File System Details**.

   The Console displays metadata for the file system, exports and snapshots for the file system, and status for the file system and its exports in associated mount targets.

To change the file system name

You can change the display name of the file system.

1. Open the navigation menu. Under **Core Infrastructure**, click **File Storage** and then click **File Systems**.
2. In the **List Scope** section, select a compartment.
3. To view information about a file system, find the file system, click the Actions icon (three dots), and then click **View File System Details**.
4. Click **Rename**.
5. Enter the new file system name, and click **Rename**.

To create an export for a file system

Exports control how NFS clients access file systems when they connect to a mount target. File systems are exported (made available) through mount targets. Each mount target maintains an export set which contains one or many exports. A file system may be exported through one

or more mount targets. A file system must have at least one export in one mount target in order for instances to mount the file system. The information used by an export includes the file system OCID, mount target OCID, export set OCID, export path, and client export options. Typically, an export is created in a mount target when the file system is created. Thereafter, you can create additional exports for a file system in any mount target that resides in the same availability domain as the file system.

1. Open the navigation menu. Under **Core Infrastructure**, click **File Storage** and then click **File Systems**.
2. In the left-hand navigation, in the **List Scope** section, under **Compartment**, select a compartment.
3. Click the name of the file system you want to create an export for, and click **Create Export**.



Note

File systems are encrypted by default. You cannot turn off encryption.

4. You can choose to accept the system defaults, or change them by clicking **Edit Details**.
5. If you want to *accept the defaults* for the mount target, click **Create**. *The file system is created with the information displayed*. If you want to choose another mount target or change the default information, click the **Edit Details** link.
6. In the **Mount Target Information** section, specify details for the mount target that is associated with the file system:
    - **Select an Existing Mount Target:** Choose this option if you want to associate the file system with a mount target you already created. Choose the **Mount Target** from the list. Click the **click here** link in the dialog box if you want to enable compartment selection for the mount target.



Tip

If there aren't any mount targets in the current combination of availability domain and compartment, this option is disabled. You can:

- Choose a different compartment.
- Create a new mount target.

- **Create a New Mount Target:** Choose this option if you want to create a new mount target associated with this file system. By default, the mount target is created in your current compartment and you can use network resources in that compartment. Click the **click here** link in the dialog box if you want to enable compartment selection for the mount target, its VCN, or subnet resources.



Important

The mount target is always in the same availability domain as the file system. While it is possible to access mount targets from any AD in a region, for optimal performance, your mount target and file system should be in the same availability domain as the Compute instances that access them. For more information, see Regions and Availability Domains.

- **Create in Compartment:** Specify the compartment you want to create the mount target in.
- **New Mount Target Name:** Optionally, replace the default with a friendly name

for the mount target. It doesn't have to be unique; an Oracle Cloud Identifier (OCID) uniquely identifies the mount target. Avoid entering confidential information.



Note

The mount target name is different than the DNS hostname, which is specified in step 7.

- **Virtual Cloud Network Compartment:** The compartment containing the cloud network (VCN) in which to create the mount target.
- **Virtual Cloud Network:** Select the cloud network (VCN) where you want to create the new mount target.
- **Subnet Compartment:** Specify the compartment containing a subnet within the VCN to attach the mount target to.
- **Subnet:** Select a subnet to attach the mount target to. Subnets can be either AD-specific or regional (regional ones have "*regional*" after the name). For more information, see About Regional Subnets.



Warning

Each mount target requires three internal IP addresses in the subnet to function. Do not use /30 or smaller subnets for mount target creation because they do not have sufficient available IP addresses. Two of the IP addresses are used during mount target creation. The third IP address must remain available for the mount target to use for high availability failover.

7. Optionally, click **Show Advanced Options** to configure the mount target's advanced options.
    - **IP Address:** You can specify an unused IP address in the subnet you selected for the mount target.
    - **Hostname:** You can specify a hostname you want to assign to the mount target.



Note

The File Storage service constructs a fully qualified domain name (FQDN) by combining the hostname with the FQDN of the subnet the mount target is located in.

For example, `myhostname.subnet123.dnslabel.oraclevcn.com`.

Once created, the hostname may be changed in the mount target's Details page. See Managing Mount Targets for more information.

8. Click **Create**.

Next, mount the file system from an instance so that you can read and write directories and files in your file system. See Mounting File Systems for instructions about obtaining mount commands for your operating system type and mounting your file system.

To set the file system reported size

The File Storage service reports file system capacity as 8589934592 gibibytes (GiB) and 8589934592 gibiinodes (GiI) by default. Sometimes, application installers perform a space requirement check prior to running an installation process but cannot correctly interpret the reported size or reported inodes of the file system. When this occurs, you can define the file system size reported to the operating system by setting the **Reported Size** or **Reported Inodes** value in the file system's mount target. Typically, setting the size to 1024 GiB and the inodes to 1024 GiI permits successful installation.



Important

Changing the **Reported Size** or **Reported Inodes** for a mount target affects all file systems exported by the mount target. **Changing these values does not limit the amount of data you can store.**

1. Open the navigation menu. Under **Core Infrastructure**, click **File Storage** and then click **File Systems**.
2. In the **List Scope** section, select a compartment.
3. Find the mount target you're interested in, click the Actions icon (three dots), and then click **View File System Details**.
4. In **Exports**, click on the mount target name.
5. Click the **Reported Size (in GiB) Edit** or the **Reported Inodes (in Gil)** icon.
6. Enter the maximum free space in gibibytes or the maximum inodes in gibinodes you want the File Storage service to report.
7. Click the **Save** icon.



Important

There can be a delay of up to 1 hour when reporting file system usage, either in the console or by using the `df` command.

To manage tags for a file system

1. Open the navigation menu. Under **Core Infrastructure**, click **File Storage** and then click **File Systems**.

2. In the **List Scope** section, select a compartment.
3. Find the file system you're interested in, click the Actions icon (three dots), and then click **View File System Details**.
4. Click the **Tags** tab to view or edit the existing tags. Or click **Apply tag(s)** to add new ones.

For more information, see Resource Tags.

To delete a file system

You can permanently delete a file system.



Warning

You cannot undo this operation. Any data in a file system is permanently deleted with the file system. Snapshots of the file system are permanently deleted with the file system. You cannot recover a deleted file system or its snapshots.

1. Open the navigation menu. Under **Core Infrastructure**, click **File Storage** and then click **File Systems**.
2. In the **List Scope** section, select a compartment.
3. Find the file system you want to delete.
4. Click the Actions icon (three dots), and then click **View File System Details**.
5. Delete all of the file system's exports:
   - In **Exports**, select the check box for all exports listed, and then click **Delete**.
6. When all of the exports are deleted, click **Delete** to delete the file system.

The file system is deleted immediately, along with all of its snapshots.

Using the Command Line Interface (CLI)

For information about using the CLI, see Command Line Interface (CLI).

To list file systems

Open a command prompt and run `oci fs file-system list` to list all the file systems in a specified availability domain and compartment.

For example:

```
oci fs file-system list --availability-domain <target_availability_domain> --compartment-id <target_compartment_id>
```

To get a specific file system

Open a command prompt and run `oci fs file-system get` to retrieve information about a specific file system.

For example:

```
oci fs file-system get --file-system-id <file_system_OCID>
```

To update a file system

Open a command prompt and run `oci fs file-system update` to update a specific file system's information.

For example:

```
oci fs file-system update --file-system-id <file_system_OCID> --display-name "<New File System Name>"
```



Warning

Avoid entering confidential information in the file system `display-name`.

To create an export for a file system

Exports control how NFS clients access file systems when they connect to a mount target. File systems are exported (made available) through mount targets. Each mount target maintains an export set which contains one or many exports. A file system may be exported through one or more mount targets. A file system must have at least one export in one mount target in order for instances to mount the file system. The information used by an export includes the file system OCID, mount target OCID, export set OCID, export path, and client export options. Typically, an export is created in a mount target when the file system is created. Thereafter, you can create additional exports for a file system in any mount target that resides in the same availability domain as the file system.

Open a command prompt and run `oci fs export create` to create an export for a specified file system within a specified export set.

For example:

```
oci fs export create --export-set-id <export_set_OCID> --file-system-id <file_system_OCID> --path
"</pathname>"
```

Important

The path must start with a slash (/) followed by a sequence of zero or more slash-separated elements. For multiple file systems associated with a single mount target, the path sequence for the first file system cannot contain the complete path element sequence of the second file system path sequence. Paths cannot end in a slash. No path element can be a period (.) or two periods in sequence (..). Lastly, no path can exceed 255 bytes. For example:

Acceptable:

`/example` and `/path`

`/example` and `/example2`

Not Acceptable:

`/example` and `/example/path`

`/` and `/example`

`/example/`

`/example/path/../example1`

Warning

If one file system associated to a mount target has '/' specified as an export path, you can't associate another file system with that mount target.



Note

Export paths cannot be edited after the export is created. If you want to use a different export path, you must create a new export with the desired path. Optionally, you can then delete the export with the old path.

For more information, see Paths in File Systems.

To set the file system reported free space

Some existing application installers perform a capacity check before running an installation process. Sometimes an installation fails because of too much available capacity. The File Storage service currently reports 8 exabytes of available capacity by default for each file system.

Customers can define how much free capacity is reported as available to the operating system.

Open a command prompt and type in the following command:

```
oci fs export-set update --export-set-id <export_set_ OCID> --max-fs-stat-bytes <number_of_bytes>
```



Important

The maximum free space setting affects each export in the export set. Setting the maximum free space does not limit the amount of data you can store.

To delete a file system

You can delete a file system if no non-deleted export resources reference it. Deleting a file system also deletes all its snapshots.

Open a command prompt and run `oci fs file-system delete` to delete a file system.

For example:

```
oci fs file-system delete --file-system-id <file_system_OCID>
```



Warning

You cannot undo this operation. Any data in a file system is permanently deleted with the file system. Snapshots of the file system are permanently deleted with the file system. You cannot recover a deleted file system or its snapshots.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operations to manage file systems:

- ListFileSystems
- GetFileSystem
- UpdateFileSystem
- DeleteFileSystem

Managing Mount Targets

This topic describes the basics of managing mount targets.

Overview

Actions you can take to manage a mount target include:

- Viewing mount target details
- Obtaining mount command samples
- Creating a new export and file system
- Editing exports and export options
- Change the reported size of exported file systems
- Deleting a mount target

You can perform most administrative tasks for your mount targets using the Console, Command Line Interface (CLI), or API. You can use the Console to list mount targets exporting a specific file system. Use the API or CLI if you want to list all mount targets in a compartment.

Mount Target

A mount target is an NFS endpoint that lives in a VCN subnet of your choice and provides network access for file systems. The mount target provides the IP address or DNS name that is used together with a unique export path to mount the file system. A single mount target can export many file systems. Typically, you create your first mount target and export when you create your first file system. The mount target maintains an export set which contains all of the exports for its associated file systems.

Exports

Exports control how NFS clients access file systems when they connect to a mount target. File systems are exported (made available) through mount targets. Each mount target maintains an export set which contains one or many exports. A file system may be exported through one or more mount targets. A file system must have at least one export in one mount target in order for instances to mount the file system. The information used by an export includes the file system OCID, mount target OCID, export set OCID, export path, and client export options. Typically, an export is created in a mount target when the file system is created. Thereafter,

you can create additional exports for a file system in any mount target that resides in the same availability domain as the file system.

NFS Export Options

NFS export options are a set of parameters within the export that specify the level of access granted to NFS clients when they connect to a mount target. An NFS export options entry within an export defines access for a single IP address or CIDR block range.

For more information, see Working with NFS Export Options.

Limitations and Considerations

- Each availability domain is limited to two mount targets by default. However, you can export up to 100 file systems through each mount target.

  See Service Limits for a list of applicable limits and instructions for requesting a limit increase.

- Each mount target requires three internal IP addresses in the subnet to function. Two of the IP addresses are used during mount target creation. The third IP address must remain available for the mount target to use for high availability failover.
- The File Storage service doesn't "reserve" the third IP address required for high availability failover. Use care when designing your subnets and file systems to ensure that sufficient IP addresses remain available for your mount targets.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let users create, manage, and delete file systems allows users to manage mount targets. Since mount targets are network endpoints, users must also

have "use" permissions for VNICs, private IPs, private DNS zones, and subnets to create or delete a mount target. See the Policy Reference for more information.

If you're new to policies, see Getting Started with Policies and Common Policies.

Tagging Resources

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

Details About Your Mount Target

The mount target details page provides the following information about your mount target:

MOUNT TARGET OCID

Every Oracle Cloud Infrastructure resource has an Oracle-assigned unique ID called an Oracle Cloud Identifier (OCID). You need your mount target's OCID to use the Command Line Interface (CLI) or the API. You also need the OCID when contacting support.

CREATED

The date and time that the mount target was created.

AVAILABILITY DOMAIN

When you create a mount target, you specify the availability domain that it resides in. An availability domain is one or more data centers located within a region. You need your mount target's availability domain to use the Command Line Interface (CLI) or the API. For more information, see Regions and Availability Domains.

COMPARTMENT

When you create a mount target, you specify the compartment that it resides in. A compartment is a collection of related resources (such as cloud networks, compute instances, or file systems) that are accessible only to those groups that have been given permission by an administrator in your organization. You need your mount target's

compartment to use the Command Line Interface (CLI) or the API. For more information, see Managing Compartments.

Reported Size (GiB)

The maximum capacity in gibibytes reported by the file systems exported through this mount target. The File Storage service currently reports 8589934592 gibibytes (GiB) of available capacity by default. If you are installing an application that requires a specific reported size, you can change the reported size. Typically, setting the size to 1024 GiB is sufficient for most applications. This value is updated hourly. See To set the file system reported size for more information.

Reported Inodes (GiI)

The maximum capacity in gibiinodes reported by the file systems exported through this mount target. The File Storage service currently reports gibiinodes (GiI) of available inodes by default. If you are installing an application that requires specific reported inodes, you can change the reported inodes. Typically, setting the inodes to 1024 GiI is sufficient for most applications. This value is updated hourly. See To set the file system reported size for more information.

Virtual Cloud Network

The VCN that contains the subnet where the mount target VNIC resides.

Subnet

The subnet within the VCN where the mount target VNIC resides. Subnets can be either AD-specific or regional (regional ones have "*regional*" after the name). For more information, see About Regional Subnets.

IP Address

The IP address that was assigned to the mount target when it was created. You need your mount target's IP address to mount associated file systems.

HOSTNAME

The hostname that was assigned to the mount target, if any. For more information about hostnames, see DNS in Your Virtual Cloud Network.

FULLY QUALIFIED DOMAIN NAME

The hostname together with the subnet domain name. For more information, see DNS in Your Virtual Cloud Network. If you specify a hostname, you can use the FDQN to mount the file system.

EXPORT SET OCID

The OCID of the mount target's export set resource. Each mount target has one export set, which contains all of the exports for the mount target. You need your mount target's export set OCID when you perform export-related tasks in the Command Line Interface (CLI) or the API.

EXPORTS

All of the mount target's exports are listed here. The export path and name of each file system is also listed. You need the export path to mount a file system.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Using the Console

To create a mount target



Important

While it is possible to access mount targets from any AD in a region, for optimal performance, your mount targets should be in the same availability domain as the Compute instances that access them. For more information, see Regions and Availability Domains.

1. Open the navigation menu. Under **Core Infrastructure**, click **File Storage** and then click **Mount Targets**.
2. In the **List Scope** section, select a compartment.
   The Console displays a list of mount targets that have already been created in the compartment, if any.
3. Click **Create Mount Target**.
4. Enter the required mount target information. Click the **click here** link in the dialog box if you want to enable compartment selection for the mount target, its VCN, or subnet resources:
   - **New Mount Target Name:** Optionally, replace the default with a friendly name for the mount target. It doesn't have to be unique; an Oracle Cloud Identifier (OCID) uniquely identifies the mount target. Avoid entering confidential information.



Note

The mount target name is different than the DNS hostname, which is specified in step 5.

- **Virtual Cloud Network Compartment:** The compartment containing the cloud network (VCN) in which to create the mount target.
- **Virtual Cloud Network:** Select the cloud network (VCN) where you want to create the new mount target.
- **Subnet Compartment:** Specify the compartment containing a subnet within the VCN to attach the mount target to.
- **Subnet:** Select a subnet to attach the mount target to. Subnets can be either AD-specific or regional (regional ones have "*regional*" after the name). For more information, see About Regional Subnets.



Warning

Each mount target requires three internal IP addresses in the subnet to function. Do not use /30 or smaller subnets for mount target creation because they do not have sufficient available IP addresses. Two of the IP addresses are used during mount target creation. The third IP address must remain available for the mount target to use for high availability failover.

5. Optionally, click **Show Advanced Options** to configure the mount target's advanced options.

- **IP Address:** You can specify an unused IP address in the subnet you selected for the mount target.
- **Hostname:** You can specify a hostname you want to assign to the mount target.



Note

The File Storage service constructs a fully qualified domain name (FQDN) by combining the hostname with the FQDN of the subnet the mount target is located in.

For example, `myhostname.subnet123.dnslabel.oraclevcn.com`.

Once created, the hostname may be changed in the mount target's Details page. See Managing Mount Targets for more information.

6. Click **Create**.

To view details of a mount target

1. Open the navigation menu. Under **Core Infrastructure**, click **File Storage** and then click **Mount Targets**.
2. In the **List Scope** section, select a compartment.
   The Console displays a list of mount targets that have already been created in the compartment, if any.
3. Find the mount target you're interested in, click the Actions icon (three dots), and then click **View Mount Target Details**.

To change the mount target name

You can change the display name of the mount target.



Note

Changing the display name doesn't affect mounting file systems exported through the mount target.

1. Open the navigation menu. Under **Core Infrastructure**, click **File Storage** and then click **Mount Targets**.
2. In the **List Scope** section, select a compartment.
3. To view information about a file system, find the file system, click the Actions icon (three dots), and then click **View Mount Target Details**.
4. Click **Rename**.
5. Enter the new mount target name, and click **Rename**.

To create an export and a new file system

Exports control how NFS clients access file systems when they connect to a mount target. File systems must have at least one export in at least one mount target in order for instances to mount the file system. The following steps create an export and a new file system. If you want to create an export for an *existing* file system, see To create an export for a file system.

1. Open the navigation menu. Under **Core Infrastructure**, click **File Storage** and then click **Mount Targets**.
2. In the left-hand navigation, in the **List Scope** section, under **Compartment**, select a compartment.
3. Click the name of the mount target you want to create an export for, and click **Create Export**.



Note

File systems are encrypted by default. You cannot turn off encryption.

4. You can choose to accept the system defaults, or change them by clicking **Edit Details**.
5. Click **Create**.

Next, mount the file system from an instance so that you can read and write directories and files in your file system. See Mounting File Systems for instructions about obtaining mount commands for your operating system type and mounting your file system.

To set the file system reported size

The File Storage service reports file system capacity as 8589934592 gibibytes (GiB) and 8589934592 gibiinodes (GiI) by default. Sometimes, application installers perform a space requirement check prior to running an installation process but cannot correctly interpret the reported size or reported inodes of the file system. When this occurs, you can define the file system size reported to the operating system by setting the **Reported Size** or **Reported Inodes** value in the file system's mount target. Typically, setting the size to 1024 GiB and the inodes to 1024 GiI permits successful installation.



Important

Changing the **Reported Size** or **Reported Inodes** for a mount target affects all file systems exported by the mount target. **Changing these values does not limit the amount of data you can store.**

1. Open the navigation menu. Under **Core Infrastructure**, click **File Storage** and then click **Mount Targets**.
2. In the **List Scope** section, select a compartment.
3. Find the mount target you're interested in, click the Actions icon (three dots), and then click **View Mount target Details**.
4. Click the **Reported Size (in GiB) Edit** or the **Reported Inodes (in Gil)** icon.
5. Enter the maximum size in gibibytes or the maximum inodes in gibiinodes you want the File Storage service to report.
6. Click the **Save** icon.



Important

There can be a delay of up to 1 hour when reporting file system usage, either in the console or by using the `df` command.

To delete an export



Note

Deleting an export does not impact the data stored in the associated file system. Deleting an export disconnects any instance that mounts the file system with the deleted export path. Mount targets that have no exports still count toward your service limit.

1. Open the navigation menu. Under **Core Infrastructure**, click **File Storage** and then click **Mount Targets**.

2. In the **List Scope** section, select a compartment.
3. Find the mount target you're interested in, click the Actions icon (three dots), and then click **View Mount target Details**.
4. In **Exports**, find the export you want to delete.
5. Click the Actions icon (three dots), and then click **Delete**.

To manage tags for a mount target

1. Open the navigation menu. Under **Core Infrastructure**, click **File Storage** and then click **Mount Targets**.
2. In the **List Scope** section, select a compartment.
3. Find the mount target you're interested in, click the Actions icon (three dots), and then click **View Mount Target Details**.
4. Click the **Tags** tab to view or edit the existing tags. Or click **Apply tag(s)** to add new ones.

For more information, see Resource Tags.

To delete a mount target

1. Open the navigation menu. Under **Core Infrastructure**, click **File Storage** and then click **Mount Targets**.
2. In the **List Scope** section, select a compartment.
3. Find the mount target you want to delete.
4. Click the Actions icon (three dots), and then click **Delete**.



Warning

Deleting the mount target also deletes all of its exports of associated file systems. File systems are no longer available through the deleted mount target.

Deleting a mount target has no effect on file system data or file system snapshots.

Using the Command Line

For information about using the CLI, see Command Line Interface (CLI).

To create a mount target

You can create a mount target for file systems in a specified compartment and subnet. A file system can only be associated with a mount target in the same availability domain.



Warning

Each mount target requires three internal IP addresses in the subnet to function. Do not use /30 or smaller subnets for mount target creation because they do not have sufficient available IP addresses. Two of the IP addresses are used during mount target creation. The third IP address must remain available for the mount target to use for high availability failover.

Open a command prompt and run `oci fs mount-target create` to create a mount target.

For example:

```
oci fs mount-target create --availability-domain <target_availability domain> --compartment-id <target_compartment_id> --subnet-id <subnet_OCID> --display-name “<My Mount Target>”
```



Warning

Avoid entering confidential information in the mount target `display-name`.

To update a mount target

Open a command prompt and run `oci fs mount-target update` to update a specific mount target's information.

For example:

```
oci fs mount-target update --mount-target-id <mount_target_OCID> --display-name "<New Mount Target Name>"
```



Warning

Avoid entering confidential information in the mount target `display-name`.

To delete a mount target

Open a command prompt and run `oci fs mount-target delete` to delete a mount target. Deleting a mount target also deletes the mount target's VNICs.

For example:

```
oci fs mount-target delete --mount-target-id <mount_target_OCID>
```



Warning

Deleting a mount target can cause any clients that have mounted an associated file system to hang. Be sure to have all clients unmount the file systems before deleting the mount target.

To list mount targets

You cannot use the Console to list mount targets. Use the command line interface or the API from a host machine running a UNIX-style operating system.

Open a command prompt and run `oci fs mount-target list` to list all mount targets in a specified availability domain and compartment.

For example:

```
oci fs mount-target list --availability-domain <target_availability_domain> --compartment-id <target_compartment_OCID>
```

To get a specific mount target

Open a command prompt and run `oci fs mount-target get` to retrieve information about a specific mount target.

For example:

```
oci fs mount-target get --mount-target-id <mount_target_OCID>
```

To create an export

Exports control how NFS clients access file systems when they connect to a mount target. File systems are exported (made available) through mount targets. Each mount target maintains an export set which contains one or many exports. A file system may be exported through one

or more mount targets. A file system must have at least one export in one mount target in order for instances to mount the file system. The information used by an export includes the file system OCID, mount target OCID, export set OCID, export path, and client export options. Typically, an export is created in a mount target when the file system is created. Thereafter, you can create additional exports for a file system in any mount target that resides in the same availability domain as the file system.

Open a command prompt and run `oci fs export create` to create an export for a specified file system within a specified export set.

For example:

```
oci fs export create --export-set-id <export_set_OCID> --file-system-id <file_system_OCID> --path
"</pathname>"
```

Important

The path must start with a slash (/) followed by a sequence of zero or more slash-separated elements. For multiple file systems associated with a single mount target, the path sequence for the first file system cannot contain the complete path element sequence of the second file system path sequence. Paths cannot end in a slash. No path element can be a period (.) or two periods in sequence (..). Lastly, no path can exceed 255 bytes. For example:

Acceptable:

`/example` and `/path`

`/example` and `/example2`

Not Acceptable:

`/example` and `/example/path`

`/` and `/example`

`/example/`

`/example/path/../example1`

Warning

If one file system associated to a mount target has '/' specified as an export path, you can't associate another file system with that mount target.



Note

Export paths cannot be edited after the export is created. If you want to use a different export path, you must create a new export with the desired path. Optionally, you can then delete the export with the old path.

For more information, see Paths in File Systems.

To list exports

Open a command prompt and run `oci fs export list` to list all exports in a specified compartment.

For example:

```
oci fs export list --compartment-id <target_compartment_id>
```

To get a specific export

Open a command prompt and run `oci fs export get` to retrieve information about a specific export.

For example:

```
oci fs export get --export-id <export_OCID>
```

To delete an export

Open a command prompt and run `oci fs export delete` to delete an export.

For example:

```
oci fs export delete --export-id <export_OCID>
```



Warning

When you delete an export, any file system referenced by the export is no longer accessible through the associated mount target.

To list export sets

Open a command prompt and run `oci fs export-set list` to list all export sets in a specified availability domain and compartment.

For example:

```
oci fs export-set list --availability-domain <target_availability_domain> --compartment-id <target_compartment_OCID>
```

To get a specific export set

Open a command prompt and run `oci fs export-set get` to retrieve information about a specific export set.

For example:

```
oci fs export-set get --export-set-id <export_set_OCID>
```

To update an export set

Open a command prompt and run `oci fs export-set update` to update a specific export set's information.

For example:

```
oci fs export-set update --export-set-id <export_set_OCID> --display-name "<New Export Set Name>"
```

To set the file system reported size

The File Storage service reports file system capacity as 8589934592 gibibytes (GiB) and 8589934592 gibiinodes (GiI) by default. Sometimes, application installers perform a space requirement check prior to running an installation process but cannot correctly interpret the reported size or reported inodes of the file system. When this occurs, you can define the file system size reported to the operating system by setting the **Reported Size** or **Reported Inodes** value in the export set of the file system's mount target. Typically, setting the size to 1024 GiB and the inodes to 1024 GiI permits successful installation.



Important

Changing the **Reported Size** or **Reported Inodes** for a mount target affects all file systems exported by the mount target. **Changing these values does not limit the amount of data you can store.**



Important

There can be a delay of up to 1 hour when reporting file system usage, either in the console or by using the `df` command.

Open a command prompt and type in the following command:

```
oci fs export-set update --export-set-id <export_set_ OCID> --max-fs-stat-bytes <number_of_bytes>
```

Using the API

- CreateMountTarget
- UpdateMountTarget
- DeleteMountTarget

- GetMountTarget
- ListMountTargets
- CreateExport
- DeleteExport
- GetExport
- ListExports
- UpdateExportSet
- GetExportSet
- ListExportSets

Working with NFS Export Options

This topic describes the basic features of NFS export options, and how to control client access to your file system.

Overview

NFS export options enable you to create more granular access control than is possible using just security list rules to limit VCN access. You can use NFS export options to specify access levels for IP addresses or CIDR blocks connecting to file systems through exports in a mount target. Access can be restricted so that each client's file system is inaccessible and invisible to the other, providing better security controls in multi-tenant environments.

Using NFS export option access controls, you can limit clients' ability to connect to the file system and view or write data. For example, if you want to allow clients to consume but not update resources in your file system, you can set access to Read Only. You can also reduce client root access to your file systems and map specified User IDs (UIDs) and Group IDs (GIDs) to an anonymous UID/GID of your choice. For more information about how NFS export options work with other security layers, see About Security.



Tip

Watch a video about working with NFS export options in File Storage.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let users create, manage, and delete file systems allows users to manage NFS export options.

If you're new to policies, see Getting Started with Policies and Common Policies.

Exports

Exports control how NFS clients access file systems when they connect to a mount target. File systems are exported (made available) through mount targets. Each mount target maintains an export set which contains one or many exports. A file system may be exported through one or more mount targets. A file system must have at least one export in one mount target in order for instances to mount the file system. The information used by an export includes the file system OCID, mount target OCID, export set OCID, export path, and client export options. Typically, an export is created in a mount target when the file system is created. Thereafter, you can create additional exports for a file system in any mount target that resides in the same availability domain as the file system.

See To create an export for a file system for more information.

NFS Export Options

NFS export options are a set of parameters within the export that specify the level of access granted to NFS clients when they connect to a mount target. An NFS export options entry within an export defines access for a single IP address or CIDR block range.

Each separate client IP address or CIDR block you want to define access for needs a separate export options entry in the export. For example, if you want to set options for NFS client IP addresses 10.0.0.6, 10.0.08, and 10.0.0.10, you need to create three separate entries, one for each IP address.

File Storage service considers the listed order of each export options entry for the export. During an NFS request by a client, File Storage service applies the first set of options that matches the client Source IP address. Only the first set is applied; the rest are ignored.

For example, consider the following two export options entries specifying access for an export:

Entry 1: Source: 10.0.0.0/16, Access: Read Only

Entry 2: Source: 10.0.0.8, Access: Read/Write

In this case, clients who connect to the export from IP address 10.0.0.8 have Read Only access. The request Source IP address is contained in the CIDR block specified in the first entry, and File Storage Service applies the options in the first match.



Important

File systems can be associated with one or more exports, contained within one or more mount targets. If the client **source** IP address does not match any entry on the list for a single export, then that export is not visible to the client. However, the file system could be accessed through other exports on the same or other mount targets. **To completely deny client access to a file system, be sure that the client source IP address or CIDR block is not included in any export for any mount target associated with the file system.**

The following options can be set to control export access:

- **Source**: The IP address or CIDR block of a connecting NFS client.
- **Require Privileged Source Port (true/false):** This setting determines whether the NFS clients specified in **source** are required to connect from a privileged source port. Privileged ports are any port including 1-1023. On Unix-like systems, only the root user can open privileged ports. Setting this value to **true** disallows requests from unprivileged ports. The default for this setting is different depending on how the export is created. Creating an export without an explicit `ClientOption` array sets the `requirePrivilegedSourcePort` attribute of the client option to **false**. When you create a `ClientOption` array explicitly , `requirePrivilegedSourcePort` defaults to **true**. For example, creating an export in the Console using the default selections sets `requirePrivilegedSourcePort` to **false**. Creating an export in the API along with a `ClientOption` array sets `requirePrivilegedSourcePort` to **true**.



Important

When **Require Privileged Source Port** is set to **true**, you also have to follow these additional configuration steps:

1. When mounting the file system from a Unix-like system, include the `resvport` option in your mount command when mounting. For example:

   ```
   sudo mount -o resvport 10.x.x.x:/fs-export-path
   /mnt/yourmountpoint
   ```

   For more information, see Mounting File Systems From Unix-Style Instances.
2. When mounting the file system from a Windows system, be sure the **UseReserverdPort**s registry key value is set to **1**.
   For more information, see Mounting File Systems From Windows Instances.

- **Access (Read_Only, Read_Write):** This setting specifies the **source** NFS client access. If unspecified, defaults to **Read_Write**.
- **Identity Squash: (All, Root, None):** This setting determines whether the **source** clients accessing the file system have their User ID (UID) and Group ID (GID) remapped to **anonymousUid** and **anonymousGid**. If you choose **All**, all users and groups are remapped. If **Root**, only the root user UID/GID combination 0/0 is remapped. If **None**, no users are remapped. If unspecified, defaults to **None**.
- **anonymousUid:** This setting is used along with the **Identity Squash** option. When remapping users, you can use this setting to change the default anonymousUid of **65534** to any user ID of your choice.

- **anonymousGid**: This setting is used along with the **Identity Squash** option. When remapping groups, you can use this setting to change the default anonymousGid of **65534** to any group ID of your choice.

Typical Access Control Scenarios

When you create file system and export, the NFS export options for that file system are set to the following defaults, which allow full access for all NFS client source connections. These defaults must be changed if you want to restrict access:

- **Source:** 0.0.0.0/0 (All)
- **Require Privileged Source Port:** False
- **Access:** Read_Write
- **Identity Squash:** None

Scenario A: Control Host Based Access

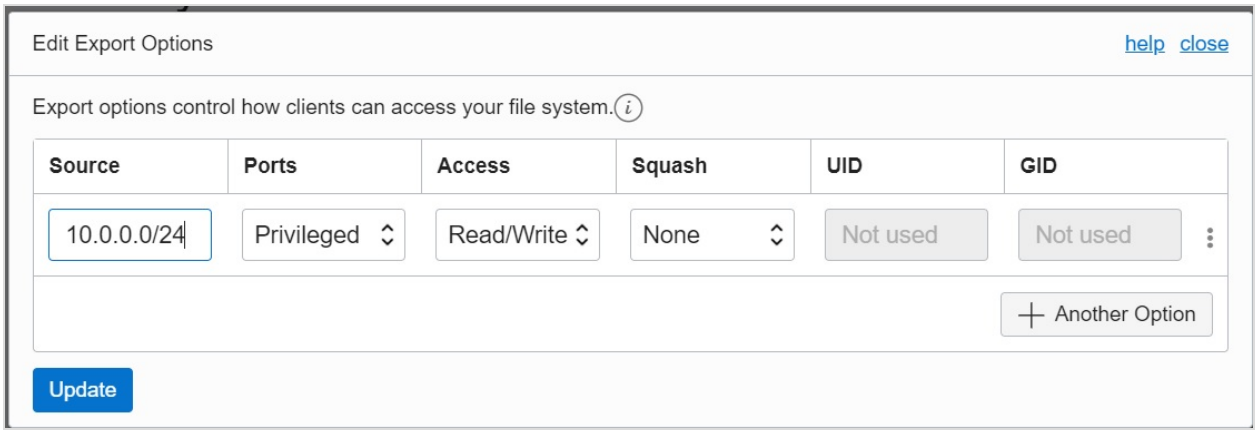
Provide a managed hosted environment for two clients. The clients share a mount target, but each has their own file system, and cannot access each other's data. For example:

- Client A, who is assigned to CIDR block 10.0.0.0/24, requires Read/Write access to file system A, but not file system B.
- Client B, who is assigned to CIDR block 10.1.1.0/24, requires Read/Write access to file system B, but not file system A.
- Client C, who is assigned to CIDR block 10.2.2.0/24, has no access of any kind to file system A or file system B.
- Both file systems A and B are associated to a single mount target, MT1. Each file system has an export contained in the export set of MT1.

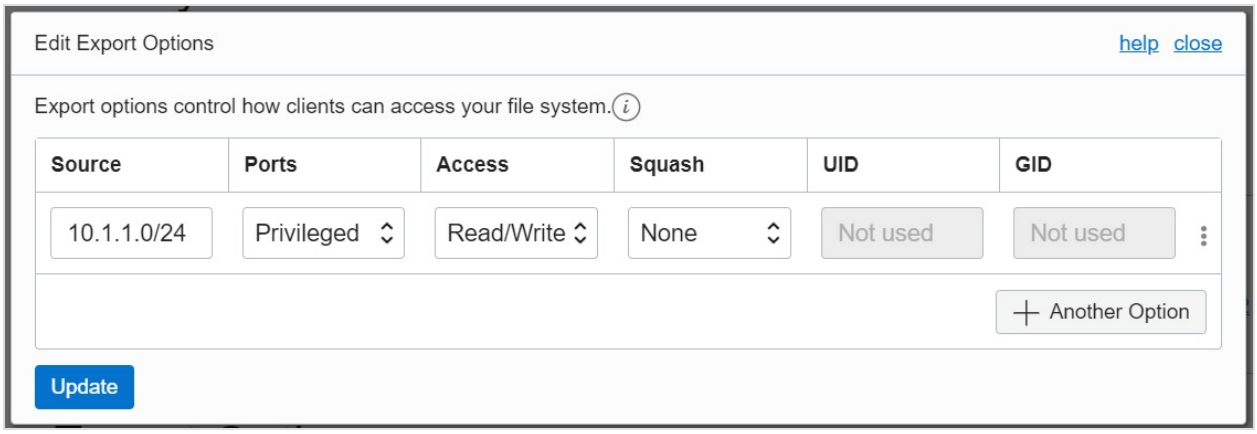
Since Client A and Client B access the mount target from different CIDR blocks, you can set the client options for both file system exports to allow access to only a single CIDR block. Client C is denied access by not including its IP address or CIDR block in the NFS export options for any export of either file system.

Console Example

Set the export options for file system A to allow Read/Write access only to Client A, who is assigned to CIDR block 10.0.0.0/24. Client B and Client C are not included in this CIDR block, and cannot access the file system.



Set the export options for file system B to allow Read/Write access only to Client B, who is assigned to CIDR block 10.1.1.0/24. Client A and Client C are not included in this CIDR block, and cannot access the file system.



CLI Example

Set the export options for file system A to allow Read_Write access only to Client A, who is assigned to CIDR block 10.0.0.0/24. Client B and Client C are not included in this CIDR block, and cannot access the file system.

```
oci fs export update --export-id <File_system_A_export_ID> --export-options '
[{"source":"10.0.0.0/24","require-privileged-source-port":"true","access":"READ_WRITE","identity-
squash":"NONE","anonymous-uid":"65534","anonymous-gid":"65534"}]'
```

Set the export options for file system B to allow Read_Write access only to Client B, who is assigned to CIDR block 10.1.1.0/24. Client A and Client C are not included in this CIDR block, and cannot access the file system.

```
oci fs export update --export-id <File_system_B_export_ID> --export-options '[{"source":"10.1.1.0/24
","require-privileged-source-port":"true","access":"READ_WRITE","identity-squash":"NONE","anonymous-
uid":"65534","anonymous-gid":"65534"}]'
```

API Example

Set the export options for file system A to allow READ_WRITE access only to Client A, who is assigned to CIDR block 10.0.0.0/24. Client B and Client C are not included in this CIDR block, and cannot access the file system.

```
PUT/<Current_API_Version>/exports/<File_System_A_export_OCID>
Host: filestorage.us-phoenix-1.oraclecloud.com
<authorization and other headers>
{
    "exportOptions": [
        {
          "source": "10.0.0.0/24",
          "requirePrivilegedSourcePort": true,
          "access": "READ_WRITE",
          "identitySquash": "NONE",
          "anonymousUid": 65534,
          "anonymousGid": 65534
        }
    ]
}
```

Set the export options for file system B to allow READ_WRITE access only to Client B, who is assigned to CIDR block 10.1.1.0/24. Client A and Client C are not included in this CIDR block, and cannot access the file system.

```
PUT/<Current_API_Version>/exports/<File_System_B_export_OCID>
Host: filestorage.us-phoenix-1.oraclecloud.com
<authorization and other headers>
{
    "exportOptions": [
        {
          "source": "10.1.1.0/24",
          "requirePrivilegedSourcePort": true,
          "access": "READ_WRITE",
          "identitySquash": "NONE",
          "anonymousUid": 65534,
          "anonymousGid": 65534
        }
    ]
}
```

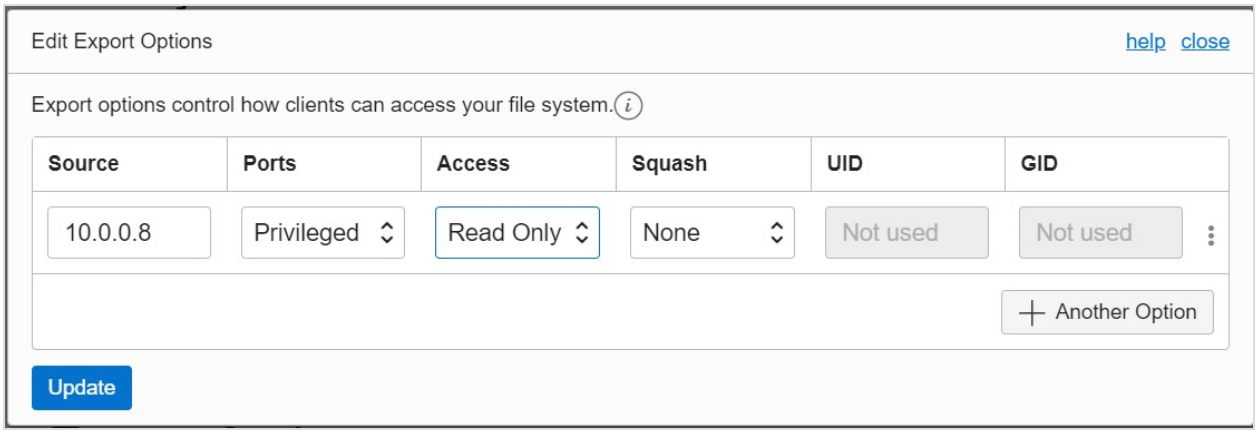
Scenario B: Limit the Ability to Write Data

Provide data to customers for consumption, but don't allow them to update the data.

For example, you'd like to publish a set of resources in file system A for an application to consume, but not change. The application connects from IP address 10.0.0.8.

Console Example

Set the source IP address 10.0.0.8 to Read Only in the export for file system A:



CLI Example

Set the source IP address 10.0.0.8 to READ_ONLY in the export for file system A:

```
oci fs export update --export-id <File_System_A_export_OCID> --export-options '
[{"source":"10.0.0.8","require-privileged-source-port":"true","access":"READ_
ONLY","identitysquash":"NONE","anonymousuid":"65534","anonymousgid":"65534"}]'
```

API Example

Set the source IP address 10.0.0.8 to READ_ONLY in the export for file system A:

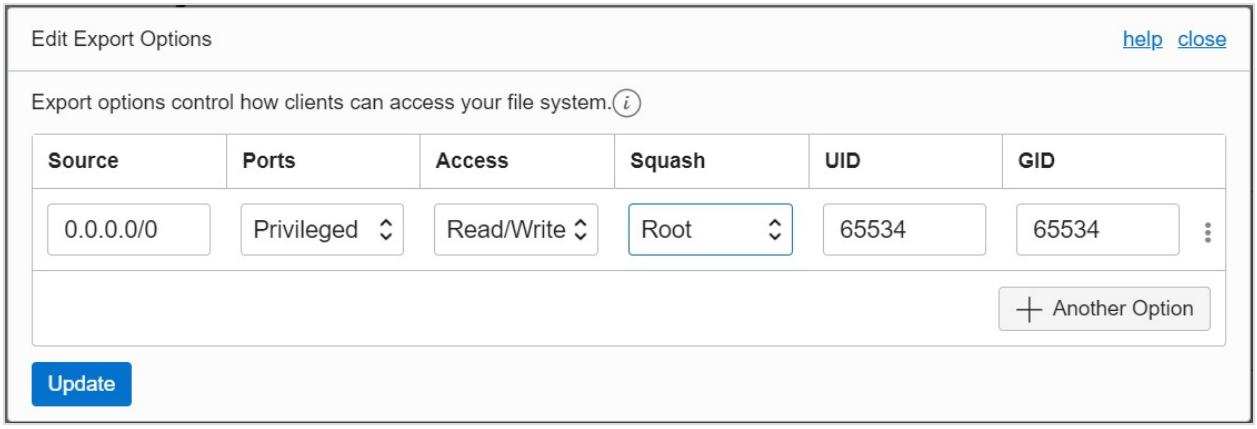
```
PUT/<Current_API_Version>/exports/<File_System_A_export_OCID>
Host: filestorage.us-phoenix-1.oraclecloud.com
<authorization and other headers>
{
    "exportOptions": [
        {
          "source": "10.0.0.8",
          "requirePrivilegedSourcePort": true,
```

```
          "access": "READ_ONLY",
          "identitySquash": "NONE",
          "anonymousUid": 65534,
          "anonymousGid": 65534
        }
    ]
}
```

SCENARIO C: IMPROVE FILE SYSTEM SECURITY

To increase security, you'd like to limit the root user's privileges when connecting to File System A. Use Identity Squash to remap root users to UID/GID 65534. In Unix-like systems, this UID/GID combination is reserved for '*nobody*', a user with no system privileges.

Console Example



CLI Example

```
oci fs export update --export-id <File_System_A_export_OCID> --export-options '
[{"source":"0.0.0.0/0","require-privileged-source-port":"true","access":"READ_
WRITE","identitysquash":"ROOT","anonymousuid":"65534","anonymousgid":"65534"}]'
```

API Example

```
PUT/<Current_API_Version>/exports/<File_System_A_export_OCID>
Host: filestorage.us-phoenix-1.oraclecloud.com
<authorization and other headers>
{
    "exportOptions": [
        {
          "source": "0.0.0.0/0",
          "requirePrivilegedSourcePort": true,
          "access": "READ_WRITE",
          "identitySquash": "ROOT",
          "anonymousUid": 65534,
          "anonymousGid": 65534
        }
    ]
}
```



Tip

If you don't want a file system to be visible to any clients, you can set all of the properties in the `exportOptions` array to empty values. For example,

```
{
   "exportOptions": [
     {
      "source":"",
      "requirePrivilegedSourcePort":"",
      "access": "",
      "identitySquash":""}
]
```

Using the Console

To set export options for a file system

1. Open the navigation menu. Under **Core Infrastructure**, click **File Storage** and then click **File Systems**.
2. In the **List Scope** section, select a compartment. All of the file systems in the selected compartment are displayed.
3. Find the file system you want to set export options for, click the the Actions icon (three dots), and then click **View File System Details**.
4. In the **Exports** list, find the export you want to set export options in, click the the Actions icon (three dots), and then click **View Export Details**. If there is no export listed for the file system, you can create one. See To create an export for a file system for more information.



Tip

To be sure you be sure that you select export, check the following:

- **The export path:** This path uniquely identifies the file system within the mount target. No two exports in a mount target can have the same export path, even if the exports are for the same file system.
- **The mount target name:** File systems can be exported through more than one mount target. Be sure that you've selected the export for the correct mount target.

5. Click **Edit Export Options**.
6. Make one or more of these changes:

- Change an export option entry in the list.
- Click **+Another Option** to create a new export option entry.
- Click the Actions icon (three dots) for an entry and move it up or down in the list.

7. When you're done, click **Update**.

Using the CLI

For information about using the CLI, see Command Line Interface (CLI).

To create an export

Open a command prompt and run `oci fs export create` to create an export for a specified file system within a specified export set. This example creates an export along with its NFS export options.

For example:

```
oci fs export create --export-set-id <export_set_OCID> --file-system-id <file_system_OCID> --path
"</pathname>" --export-options '
[{"source":"10.0.0.0/16","requireprivilegedsourceport":"true","access":"READWRITE","identitysquash":"NO
NE","anonymousuid":"0","anonymousgid":"0"}]'
```



Important

Export Path Names

The path must start with a slash (/) followed by a sequence of zero or more slash-separated elements. For any two export resources associated with the same export set, the path sequence for the first export resource can’t contain the complete path element sequence of the second export sequence. Paths can't end in a slash. No path element can be a period (.) or two periods in sequence (..). Lastly, no path can exceed 255 bytes.

Examples:

Acceptable:

`/example` and `/path`

`/example1` and `/example2`

Not Acceptable:

`/example` and `/example/path`

`/` and `/example`

`/example/`

`/example/path/../example1`

To update export options

Open a command prompt and run `oci fs export update`. To update export options for a specified file system, use `--export-options`.

For example:

```
oci fs export update --export-id <export_OCID> --export-options '[{"source":"<0.0.0.0/0>","require-privileged-source-port":"true","access":"READ_ONLY","identity-squash":"ROOT","anonymous-uid":"65534","anonymous-gid":"65534"}]'

WARNING: Updates to export-options will replace any existing values. Are you sure you want to continue? [y/N]: y
```



Tip

If you don't want a file system to be visible to any clients, you can set all of the properties in Client Options to empty values. For example,

```
oci fs export update --export-id <export_OCID> --export-options '[{"source":"","require-privileged-source-port":"true","access":"READ_ONLY","identity-squash":"ROOT","anonymous-uid":"65534","anonymous-gid":"65534"}]'
```

To list exports

Open a command prompt and run `oci fs export list` to list all exports in a specified compartment.

For example:

```
oci fs export list --compartment-id target_compartment_id
```

To delete an export

Open a command prompt and run `oci fs export delete` to delete an export.

For example:

```
oci fs export delete --export-id export_OCID
```



Warning

When you delete an export, you can no longer mount the file system using the file path specified in the deleted export.

Using the API

- CreateExport
- UpdateExport
- ListExports
- GetExport
- DeleteExport

Managing Snapshots

The File Storage service supports snapshots for data protection of your file system. Snapshots are a consistent, point-in-time view of your file systems. Snapshots are copy-on-write, and scoped to the entire file system. You can take as many snapshots as you need. Data usage is metered against differentiated snapshot data. If nothing has changed within the file system since the last snapshot was taken, the new snapshot does not consume more storage.

Snapshots are accessible under the root directory of the file system at `.snapshot/name`. For data protection, you can use rsync, tar, or another third-party tool that supports NFSv3 to copy your data to a remote location, file system, or object storage.



Tip

Watch a video about protecting data with snapshots in File Storage.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Required IAM Service Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let users create, manage, and delete file systems allows users to create and delete snapshots.

If you're new to policies, see Getting Started with Policies and Common Policies.

Tagging Resources

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

Using the Console

To create a snapshot

1. Open the navigation menu. Under **Core Infrastructure**, click **File Storage** and then click **File Systems**.
2. In the **List Scope** section, select a compartment.
3. In the **File Systems** list, locate the file system you want to take a snapshot of. Click the Actions icon (three dots), and then click **View File System Details**.
4. In **Resources**, click **Snapshots**.
5. Click **Create Snapshot**.
6. Fill out the required information:
   - **Name**: Enter a name for the snapshot. It must be unique among all other snapshots for this file system. The name can't be changed. Avoid entering confidential information.
7. Click **Create Snapshot**.

   The snapshot is accessible under the root directory of the file system at `.snapshot/`*`name`*.

To view details of a snapshot

1. Open the navigation menu. Under **Core Infrastructure**, click **File Storage** and then click **File Systems**.
2. In the **List Scope** section, select a compartment.
3. In the **File Systems** list, locate the file system you took the snapshot of. Click the Actions icon (three dots), and then click **View File System Details**.
4. In **Resources**, click **Snapshots**.

5. In the **Snapshots** list, locate the snapshot you're interested in. Click the Actions icon (three dots), and then click **View Snapshot Details**.

To create a snapshot from a Unix-style instance

You can create a snapshot from an instance that you've mounted the file system to. Snapshots are created under the root folder of your file system, in a hidden directory named `.snapshot`.

1. Connect to your instance and open a command window.
2. Navigate to your file system's hiden `.snapshot` directory. Type the following, replacing `yourmountpoint` with the name of the directory where you mounted the file system.

   ```
   cd /mnt/yourmountpoint/.snapshot
   ```

3. Use the `mkdir` command to create a directory in the hidden `.snapshot` directory. The directory you create is the snapshot. Give the snapshot a name that will help you identify it. Avoid using confidential information in the snapshot name. Avoid entering confidential information. For example:

   ```
   mkdir snapshot-Jan1
   ```

4. Use the `ls` command to verify that your snapshot has been created in the `.snapshot` directory.

   ```
   ls
   ```

To restore a snapshot

Snapshots are created under the root folder of your file system, in a hidden directory named `.snapshot`.

You can restore a file within the snapshot, or an entire snapshot using the `cp` command. Use the `-r` option when restoring a snapshot that contains subdirectories.

For example:

```
cp -r .snapshot/snapshot_name/* destination_directory_name
```

Optionally, you can use rsync, tar, or another third-party tool that supports NFSv3 to copy your data to another remote location.

To manage tags for a snapshot

1. Open the navigation menu. Under **Core Infrastructure**, click **File Storage** and then click **File Systems**.
2. In the **List Scope** section, select a compartment.
3. In the **File Systems** list, locate the file system you took the snapshot of. Click the Actions icon (three dots), and then click **View File System Details**.
4. In **Resources**, click **Snapshots**.
5. In the **Snapshots** list, locate the snapshot you're interested in. Click the Actions icon (three dots), and then click **View Snapshot Details**.
6. Click the **Tags** tab to view or edit the existing tags. Or click **Apply tag(s)** to add new ones.

For more information, see Resource Tags.

To delete a snapshot

1. Open the navigation menu. Under **Core Infrastructure**, click **File Storage** and then click **File Systems**.
2. In the **List Scope** section, select a compartment.
3. Find the file system with the snapshot you want to delete.
4. Click the Actions icon (three dots), and then click **View File System Details**.
5. In **Resources**, click **Snapshots**.
6. Find the snapshot you want to delete.
7. Click **Delete.**

Using the Command Line Interface (CLI)

For information about using the CLI, see Command Line Interface (CLI).

To create a snapshot

You can create a snapshot of a file system. A snapshot is a point-in-time view of the file system. The snapshot is accessible at ./shapshot/*name*.

Open a command prompt and run `oci fs snapshot create` to create a snapshot of a file system.

For example:

```
oci fs snapshot create --file-system-id <file_system_OCID> --name "<January1>"
```



Warning

Avoid entering confidential information in the snapshot `name`.

To list snapshots

Open a command prompt and run `oci fs snapshot create` to list all snapshots associated with a specific file system.

For example:

```
oci fs snapshot list --file-system-id <file_system_OCID>
```

To get a specific snapshot

Open a command prompt and run `oci fs snapshot get` to retrieve information about a specific snapshot.

For example:

```
oci fs snapshot get --snapshot-id <snapshot_OCID>
```

To delete a snapshot

Open a command prompt and run `oci fs snapshot delete` to delete a snapshot.

For example:

```
oci fs snapshot delete --snapshot-id <snapshot_OCID>
```

Using the API

- CreateSnapshot
- ListSnapshots
- GetSnapshot
- DeleteSnapshot

Paths in File Systems

TheFile Storage service uses three kinds of paths :

1. **Export Paths** are part of the information contained in an export that makes a file system available through a mount target. The export path uniquely identifies the file system within the mount target, letting you associate up to 100 file systems behind a single mount target. The export path is used by an instance to mount (logically attach to) the file system. This path is unrelated to any path within the file system or the client instance. It exists solely as a way to distinguish one file system from another within a single mount target.
   In this mount command example, `10.0.0.6` is the mount target IP address.`/example/path` is the unique export path that was specified when the file system was associated with a mount target during creation.

```
sudo mount 10.0.0.6:/example/path /mnt/mountpointA
```



Important

The path must start with a slash (/) followed by a sequence of zero or more slash-separated elements. For multiple file systems associated with a single mount target, the path sequence for the first file system cannot contain the complete path element sequence of the second file system path sequence. Paths cannot end in a slash. No path element can be a period (.) or two periods in sequence (..). Lastly, no path can exceed 255 bytes. For example:

Acceptable:

`/example` and `/path`

`/example` and `/example2`

Not Acceptable:

`/example` and `/example/path`

`/` and `/example`

`/example/`

`/example/path/../example1`



Warning

If one file system associated to a mount target has '/' specified as an export path, you can't associate another file system with that mount target.



Note

Export paths cannot be edited after the export is created. If you want to use a different export path, you must create a new export with the desired path. Optionally, you can then delete the export with the old path.

See Managing Mount Targets for more information about mount targets and exports.

2. **Mount Point Paths** are paths within a client instance to a locally accessible directory to which the remote file system is mounted.
   In this mount command example, `/mnt/mountpointA` is the path to the directory on the client instance on which the external file system is mounted.
   ```
   sudo mount 10.0.0.6:/example/path /mnt/mountpointA
   ```
   See Mounting File Systems for more information.
3. **File System Paths** are paths to directories within the file system, and contain the contents of the file system. When the file system is mounted, you can create any directory structure within it you like. Snapshots of the file system can be accessed using the file system path, under the file system's root directory at `.snapshot/name`.
   The following example shows the path to a snapshot called 'January 1' when navigating from the instance:
   ```
   /mountpointA/.snapshot/January1
   ```

Troubleshooting Your File System

These topics cover some common issues you may run into and how to address them:

- Application Installation Fails Due to Too Much or Too Little Available Capacity
- Application Performance is Not as Expected
- Cannot Delete VCN - Mount Target VNIC Still Attached
- Mount Command Fails

- Mount Target Creation Fails
- Mount Target is in a Failed State
- Removing File Locks from a Host that is No Longer Available
- Showmount Command Fails
- Symbolic Links (Symlinks) Produce Errors
- Troubleshooting Windows NFS Connections

Application Installation Fails Due to Too Much or Too Little Available Capacity

Some existing application installers perform a space requirements check prior to running an installation process. Sometimes an installation fails due to too much available capacity. The File Storage service currently reports 8 gibibytes of available capacity and 8 gibinodes of available inodes by default.

You can define how much free capacity is reported as available to the operating system by setting the **Reported Size** or **Reported Inodes** in the file system's mount target.



Important

There can be a delay of up to 1 hour when reporting file system usage, either in the console or by using the `df` command.

Customers can define how much free space is reported as available to the operating system using the Console, the API or the Command Line Interface (CLI). If your application installation is failing because of too little available space, you can expand the reported available free space. If your application installation is failing because of too much reported available free space, you can reduce it.

To set the file system reported size in the Console

To set the file system reported size in the CLI

To set the reported free space in the API

You can use the UpdateExportSet operation to update the `MaxFsStatBytes`.

See REST APIs for more information.

Application Performance is Not as Expected

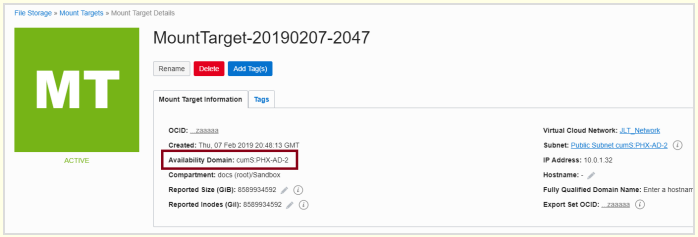
Several factors can impact application performance:

- **Available bandwidth**
  We recommend that you use bare metal Compute instances because instance bandwidth scales with the number of oCPU's. Bare metal Compute instances provide the greatest bandwidth. Virtual machines (VMs) are bandwidth limited based on the number of CPUs consumed. Single oCPU VM Compute instances provide the least bandwidth.
- **Latency**
  Subnets can be either AD-specific or regional. The type of subnet you choose to create your File Storage resources in can affect latency. You can create File Storage resources in either type of subnet.
  Regional subnets allow Compute instances to connect to any mount target in the subnet regardless of AD, with no additional routing configuration. However, to minimize latency, place mount targets in the same AD as Compute instances just as you would in an AD-specific subnet.
  For more information, see About Regional Subnets.



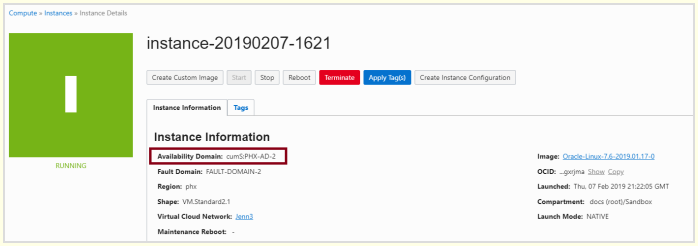
Tip

If you want to verify that your instance and mount target are in the same availability domain, you can view the availability domain for any mount target in its **Details** page, in the **Mount Target Information** tab:



A file system is always in the same subnet as its associated mount target.

You can also view the availability domain for any instance in its **Details** page, in the **Instance Information** tab:



- **Mount options**

By not providing explicit values for mount options such as rsize and wsize, the client and server can negotiate the window size for read and write operations that provide the best performance.

Cannot Delete VCN- Mount Target VNIC Still Attached

A mount target is an NFS endpoint that lives in a VCN subnet of your choice and provides network access for the file systems that it exports. Each mount target has a VNIC to enable network access. Mount target VNICs that remain in a VCN must be deleted before you can delete the VCN.

Deleting a mount target also deletes all of the exports of associated file systems that exist in its export set. Data in the file systems is not affected, but the file systems are no longer available through the deleted mount target. You can create new exports for the file system in a different mount target and subnet.

For more information, see Managing Mount Targets.

To resolve this issue using the Console

1. Note the OCID in the error message you receive when you attempt to delete the VCN. Mount target OCIDs contain the identifier `mounttarget`. For example:

   ```
   ocid1.mounttarget.oc1.phx.examplemounttargetid
   ```

2. Note the **Compartment** and **Subnet** information of the **VCN** you want to delete, and to assist navigation and choosing the correct mount target to delete.
3. Delete the mount target using the following steps:
   a. Open the navigation menu. Under **Core Infrastructure**, click **File Storage** and then click **Mount Targets**.
   b. In the **List Scope** section, select a compartment.
   c. Find the mount target you want to delete.
   d. Click the Actions icon (three dots), and then click **Delete**.



Warning

Deleting the mount target also deletes all of its exports of associated file systems. File systems are no longer available through the deleted mount target.



Tip

In the Console, the mount target OCID can be seen in the mount target details page in the **Mount Target Information** tab. See Managing Mount Targets for more information about how to view the mount target details page. Be sure the mount target OCID seen on the details page matches the mount target OCID provided by the VCN delete process error message.

4. Delete the VCN.

To resolve this issue using the API

1. Note the OCID in the error message you receive when you attempt to delete the VCN. Mount target OCIDs contain the identifier `mounttarget`. For example:

```
ocid1.mounttarget.oc1.phx.examplemounttargetid
```

2. Delete the mount target using the following steps:
   a. Use DeleteMountTarget to delete the mount target. For example:

```
DELETE /20171215/mountTargets/ocid1.mounttarget.oc1.phx.examplemounttargetid
Host: filestorage.us-phoenix-1.oraclecloud.com
<authorization and other headers
```

b. You can use GetMountTarget to verify that the mount target has been deleted. For example:

```
GET
/20171215/mountTargets/ocid1.mounttarget.oc1.phx.examplemounttargetid?compartmentId=
<compartmentId>
Host: filestorage.us-phoenix-1.oraclecloud.com
<authorization and other headers>
```

The API should return `Status 404 Not Found`.

To resolve this issue using the CLI

For general information about using the CLI, see Command Line Interface (CLI).

1. Note the OCID in the error message you receive when you attempt to delete the VCN. Mount target OCIDs contain the identifier `mounttarget`. For example:

```
ocid1.mounttarget.oc1.phx.examplemounttargetid
```

2. Delete the mount target using the following steps:

   a. Use `oci fs mount-target delete` to delete the mount target. For example:

```
oci fs mount-target delete --mount-target-id
ocid1.mounttarget.oc1.phx.examplemounttargetid
```

   b. You can use `oci fs export get` to verify that the mount target has been deleted. For example:

```
oci fs export get --export-id ocid1.mounttarget.oc1.phx.examplemounttargetid
```

   The CLI should return a message indicating the mount target is not found. For example:

```
{
    "code": "NotAuthorizedOrNotFound",
    "message": "Authorization failed or requested resource not found.",
    "opc-request-id": "<requestID>",
    "status": 404
}
```

If you still can't delete the VCN, be sure there are no other resources remaining in the VCN that might prevent it. For more information, see Subnet or VCN Deletion.

Mount Command Fails

The export path of a file system must be correctly represented in your mount command, or the mount will fail.

The export path is specified when you create an export for the file system in a mount target. It uniquely identifies the file system within the mount target, letting you associate multiple file systems to a single mount target. The export path is appended to the mount target IP address, and used to mount the file system.

```
sudo mount 10.0.0.6:/example/path /mnt/mountpointA
```

In this example, `10.0.0.6:` is the mount target IP address, and `/example/path` is the export path. `/mnt/mountpointA` is the path to the directory on the client instance on which the external file system is mounted.



Tip

You can find all the export paths for a file system in the **Exports** list shown in its Details page, together with associated mount target information.

- You can obtain the correct export path by copying mount commands directly from the file system export. These commands minimize the chance of a typing error. See To get mount command samples for more information.
- If one file system associated with a mount target uses an export path of '/' , it will prevent you from associating more file systems with that mount target. No two file systems associated with the same mount target can have an export path that contains a complete path of the other.

See Paths in File Systems for more information.

Mount Target Creation Fails

Mount target creation can fail for various reasons:

- **You've exceeded your mount target limit.**
  Each availability domain is limited to two mount targets by default. If you create both a file system and a mount target at the same time, it is possible for the file system to be successfully created but the mount target creation to fail because of this limitation.
  You can create an export for the file system in a previously existing mount target. For more information, see To create an export for a file system.
  See Service Limits for a list of applicable limits and instructions for requesting a limit increase.
- **There aren't enough available IP addresses in your subnet.**
  Each mount target requires three internal IP addresses in the subnet to function. Two of the IP addresses are used during mount target creation. The third IP address must remain available for the mount target to use for high availability failover.
  Do not use /30 or smaller subnets for mount target creation because they do not have sufficient available IP addresses for mount target creation.
  The File Storage service doesn't "reserve" the third IP address required for high availability failover, so use care when designing your subnets and file systems to ensure that sufficient IP addresses remain available for your mount targets in the future.

Mount Target is in a Failed State

Symptom: A mount target reports a **Failed** state. File systems are not accessible using the mount target's IP address.

Possible Cause: There are insufficient unallocated IP addresses in the subnet. The mount target cannot fail over successfully.

Each mount target requires three internal IP addresses in the subnet to function. Two of the IP addresses are used during mount target creation. The third IP address must remain available for the mount target to use for high availability failover. The File Storage service doesn't "reserve" the third IP address required for high availability failover. Use care to ensure that

enough unallocated IP addresses remain available for your mount targets to use during failover.

Solution:

1. Delete the failed mount target.
   To delete a mount target
2. Export the file system through an active mount target. You can create a replacement mount target and then create an export for the file system, or create an export for the file system in a pre-existing mount target.
   - You can use the same export paths for the associated file systems as the previous mount target. However, the export path must be unique for each file system within the mount target.
   - If you create a replacement mount target, you can use the same IP address as the previous mount target, if available. Be sure to explicitly specify the desired IP address when you create the mount target.

   To create a mount target
   To create an export for a file system
3. If necessary, mount the file systems again.
   Mounting File Systems

   Note

   If a replacement mount target uses *exactly the same* IP address and export paths as previously existed in the deleted mount target, mounted instances reconnect automatically.
4. To prevent a recurrence of this issue, ensure that sufficient unallocated IP addresses remain available in the subnet.

Removing File Locks from a Host that is No Longer Available

The File Storage service supports the removal of file locks from any file system. To request the removal of file locks on a file system:

1. Go to My Oracle Support and sign in.
   If you are not signed in directly to Oracle Cloud Support, click **Switch to Cloud Support** at the top of the page.
2. Click **Create Service Request**.
3. Select the following from the displayed menus:
   - **Service Type:** Select Oracle Cloud Infrastructure from the list.
   - **Service Name:** Select the appropriate option for your organization.
   - **Problem Type:** FSS File System Lock Removal Request.
4. Enter your contact information.
5. Enter a **Description**, and then enter the following required fields specific to your issue. For most Oracle Cloud Infrastructure issues you need to include the OCID (Oracle Cloud Identifier) for each resource you need help with. See Locating IDs for Your Oracle Cloud Infrastructure Resources for instructions on locating these.
   - Tenancy OCID
   - File System OCID
   - Mount Target OCID
   - Host IP Address

   For help with any of the general fields in the service request or for information on managing your service requests, click **Help** at the top of the Oracle Cloud Support page.

Showmount Command Fails

The File Storage service supports the `showmount -e` command. You can use the `showmount -e` command to show a list of NFS exports available from a specific mount target IP address.

For example, `$ showmount -e 10.0.0.0`

To enable the command, you must create a security list rule in the subnet containing the mount target. The rule must be a **stateful ingress** rule for **UDP** traffic with a **Destination Port Range** of **111**.

Click here for instructions about Configuring VCN Security List Rules for File Storage.

For more information about this command, see the Linux manual page about showmount.



Important

Only `showmount -e` is supported. No other options are supported, and the `-e` option must be included.

Symbolic Links (Symlinks) Produce Errors

The File Storage service fully supports the use of symbolic links. However, symbolic links are interpreted by the client and symlinks that point outside of the mounted File Storage system may be interpreted differently by each client and lead to unexpected results, such as broken links or pointing to the wrong file. Symbolic link targets that work on one client might be broken on another due to differences in file system layout or because clients mounted the file system using different mount targets.

Snapshots can also break symbolic links that point to a target outside the file system's root directory. This is because when you create a snapshot of a file system, it becomes available as a subdirectory of the .snapshot directory.

To minimize these potential issues, use a relative path as the target path when creating a symbolic link to a file in the network file system. Also, ensure that relative paths do not point to a target path outside the File Storage service root directory except when the target is on the local machine. If you must use a symbolic link that points to a target path outside the file system, use an absolute path starting with the client's root directory.

For example:

- Pointing to "/user/bin/example" **works**.
- Pointing to "/yourmountpoint/..." does **not** work.
- Pointing to "/home/user/yourmountpoint/..." does **not** work.

Troubleshooting Windows NFS Client Connections

This topic describes some common issues encountered when installing Windows NFS client and mounting a file system.



Important

To connect to file systems from Windows instances, the NFS Client must first be installed. Be sure to follow the installation procedure found in Mounting File Systems From Windows Instances before proceeding with troubleshooting.

Cannot Create or Write to Files on a Mounted File System

Symptom: After installing Windows NFS client, you can successfully mount the file system from Windows, but any attempt to create or update a file in the file system fails.

Cause 1: Access to NFS file systems requires UNIX-style user and group identities, which are not the same as Windows user and group identities. To enable users to access NFS shared resources, Windows client for NFS accesses file systems anonymously, using 'AnonymousGid'

and 'AnonymousUid'. On brand new file systems, write permissions are only granted to the root user.

Solution: Map the AnonymousGid and AnonymousUid to the root user, and then remount the file system with the new user privileges.

To map the AnonymousGid and AnonymousUid to the root user

1. In the Windows Command Line (CMD) window, unmount the file system by typing the following. Replace `10.x.x.x:` with the local subnet IP address assigned to your mount target, `fs-export-path` with the export path you specified when associating the file system with the mount target, and `X` with the drive letter of any available drive you want to map the file system to.

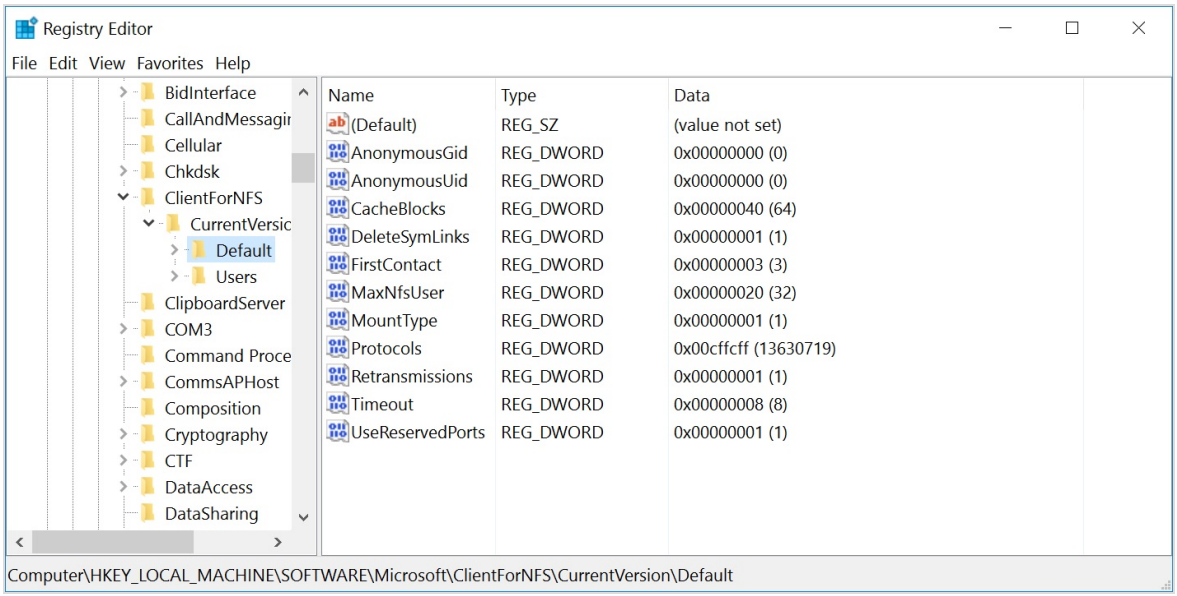
   Tip

   IP address and export path information is available in the **Details** page of the mount target associated with your file system. See To view details of a mount target for more information.

   ```
   umount 10.x.x.x:/fs-export-path X:
   ```

2. Open the registry editor (regedit):
   - Click **Windows Search.**
   - Enter `regedit` in the **Search** field and press **Enter**.
   - Click **Yes** to allow changes to your device.
3. Click `HKEY_LOCAL_MACHINE`. Then, browse to: `Software\Microsoft\ClientForNFS\CurrentVersion\Default`.
4. Add a new DWORD32 registry entry for `AnonymousGid`:
   - Click **Edit**, and select **New DWORD (32 bit) Value**.
   - In the Name field, enter `AnonymousGid`. Leave the value at `0`.

5. Repeat step 3 to add a second DWORD32 registry entry named `AnonymousUid` with a value of `0`.



6. Open Windows Command Line (CMD) and run as Administrator:
   - Go to **Start** and scroll down to **Apps**.
   - In the **Windows System** section, press **Ctrl+Shift** and click **Command Prompt**.
7. In the Windows Command Line (CMD) window, restart the NFS Client by typing the following:

```
nfsadmin client stop
```

```
nfsadmin client start
```

8. Close the Administrator: Windows Command Prompt (CMD) window. Open a **standard** Command Prompt Window:
   - Click **Start**, then click **Command Prompt**.



Important

NFS file systems mounted as Administrator are not available to standard users.

9. In the standard Windows Command Line (CMD) window, mount the file system by typing the following. Replace `10.x.x.x:` with the local subnet IP address assigned to your mount target, `fs-export-path` with the export path you specified when associating the file system with the mount target, and `X` with the drive letter of any available drive you want to map the file system to.

```
mount 10.x.x.x:/fs-export-path X:
```

Cause 2: A standard user is trying to access a file system that was mounted using the Administrator: Command Prompt (CMD). When mounting file systems, it isn't necessary to run the Command Prompt as Administrator.

Solution: Unmount the file system and then remount the file system using a standard Command Prompt. (CMD)

To remount a file system with a standard Command Prompt (CMD)

1. Open Windows Command Line (CMD) and run as Administrator:
   - Go to **Start** and scroll down to **Apps**.
   - In the **Windows System** section, press **Ctrl+Shift** and click **Command Prompt**.
2. In the Administrator: Windows Command Line (CMD) window, **unmount** the file system by typing the following. Replace `10.x.x.x:` with the local subnet IP address assigned to your mount target, `fs-export-path` with the export path you specified when associating the file system with the mount target, and `X` with the drive letter of any available drive you want to map the file system to.



Tip

IP address and export path information is available in the **Details** page of the mount target associated with your file system. See To view details of a mount target for more information.

```
umount 10.x.x.x:/fs-export-path X:
```

3. **Close** the Administrator: Windows Command Line (CMD) window.
4. Open a **standard** Command Prompt Window:
   - Click **Start**, then click **Command Prompt**.
5. In the **standard** Command Line (CMD) window, mount the file system by typing the following. Replace `10.x.x.x:` with the local subnet IP address assigned to your mount target, `fs-export-path` with the export path you specified when associating the file system with the mount target, and `X` with the drive letter of any available drive you want to map the file system to.

```
mount 10.x.x.x:/fs-export-path X:
```

Mounted Drive is Not Visible in File Explorer

Symptom: After installing Windows NFS client, you can successfully mount the file system from Windows, but the file system drive is not visible in File Explorer.

Cause: A standard user is trying to access a file system that was mounted using the Administrator: Command Prompt (CMD). When mounting file systems, it isn't necessary to run the Command Prompt as Administrator.

Solution: Unmount the file system and then remount the file system using a standard Command Prompt. (CMD) See previous information about how To remount a file system with a standard Command Prompt (CMD).

Mounting from File Explorer Fails With "An Unexpected Error Occurred."

Symptom: The IP address and export path are correctly represented in the **Folder** field. When you click **Finish**, the system attempts to connect to the file system, but fails with an error: "The mapped network drive could not be created because the following error has occurred: An unexpected error occurred."

Solution 1: Reboot the instance, and mount the file system again using File Explorer.

Solution 2: Mount the file system using the Command Prompt.

Accessing a File System Using Universal Naming Convention (UNC) Methods

You can use the following Universal Naming Convention (UNC) methods to access files in a mounted file system:

- Open the **Command Prompt** by clicking **Search** and typing `cmd`. Use the `start` command and enter the UNC path:

  ```
  start \\10.0.0.1\export_path_name\folder_or_file_name
  ```

- Open the **Command Prompt** by clicking **Search** and typing `cmd`. Use the `dir` command and enter the UNC path:

  ```
  dir \\10.0.0.1\export_path_name\folder_name
  ```

- Open the **Run** window by clicking **Search** and typing `run`. Enter the UNC path:

  ```
  \\10.0.0.1\export_path_name\folder_or_file_name
  ```

Learn more about how Windows uses the Multiple UNC Provider.

Windows 2008 R2: UNC Access Delayed; "Network Error 53 Network path not found"

Symptom1 : After installing NFS client on Windows 2008 R2 servers, mount fails with "Network Error 53 "Network path not found".

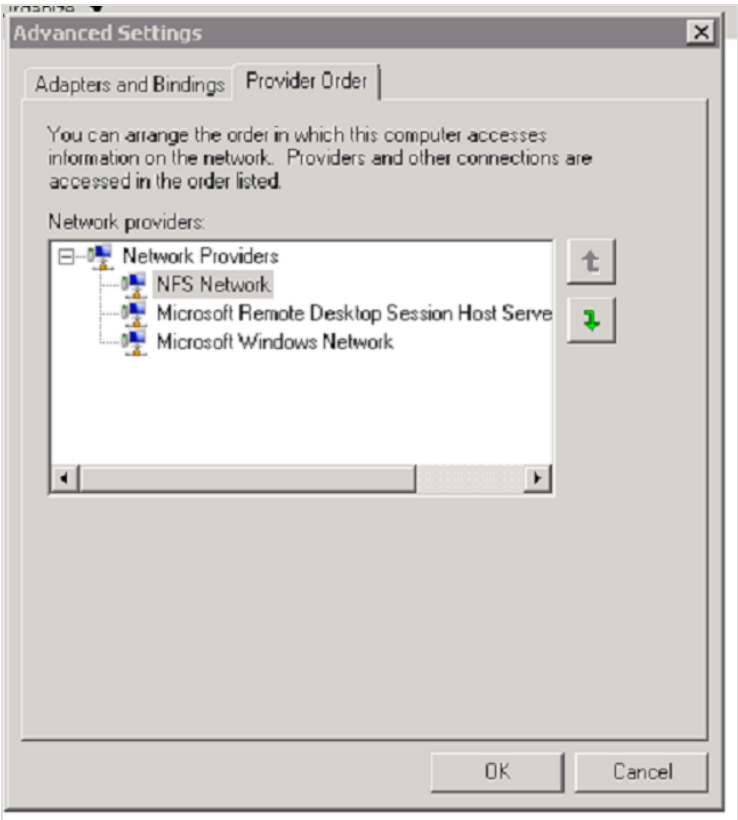
Symptom 2: After installing NFS client, connection to a file system using a Universal Naming Convention (UNC) path, is significantly delayed on Windows 2008 R2 servers.

Cause 1: Windows Network Provider has higher priority than Client for NFS Network Provider.

Solution 1: Change the Network Provider Order so that Client for NFS Network Provider is tried first.

To change the Network Provider Order on Windows 2008 R2

1. Click **Start**, and then click **Network**.
2. **Right-click** or press **Shift+F10** and click **Properties**.
3. Click **Change Adapter Settings**. Press **Alt** and click **Advanced** in the menu.
4. In the **Advanced** menu, click **Advanced Settings**.
5. Click the **Provider Order** tab.
6. Select the **NFS Network** from the list of Network providers.
7. Click the **Up Arrow** to move the NFS Network provider to the top of the list.
8. Click **OK**.



Cause 2: Mounting a file system requires stateful ingress TCP ports 111, 2048, 2049, and 2050 in addition to stateful ingress UDP ports 111 and 2048. The security list rules for the subnet where the mount target resides are not correctly set up.

Solution 2: Use the instructions in Configuring VCN Security List Rules for File Storage to set up the correct security list rules.

CHAPTER 13 IAM

This chapter explains how to set up administrators, users, and groups and specify their permissions.

Overview of Oracle Cloud Infrastructure Identity and Access Management

Oracle Cloud Infrastructure Identity and Access Management (IAM) lets you control who has access to your cloud resources. You can control what type of access a group of users have and to which specific resources. This section gives you an overview of IAM components and an example scenario to help you understand how they work together.



Note

This document uses the term "you" broadly to mean any administrator in your company who has access to work with IAM.

Components of IAM

IAM uses the components described in this section. To better understand how the components fit together, see Example Scenario.

RESOURCE

The cloud objects that your company's employees create and use when interacting with Oracle Cloud Infrastructure. For example: compute instances, block storage volumes, virtual cloud networks (VCNs), subnets, route tables, etc.

USER

An individual employee or system that needs to manage or use your company's Oracle Cloud Infrastructure resources. Users might need to launch instances, manage remote disks, work with your virtual cloud network, etc. End users of your application are not typically IAM users. Users have one or more IAM credentials (see User Credentials).

GROUP

A collection of users who all need the same type of access to a particular set of resources or compartment.

DYNAMIC GROUP

A special type of group that contains resources (such as compute instances) that match rules that you define (thus the membership can change dynamically as matching resources are created or deleted). These instances act as "principal" actors and can make API calls to services according to policies that you write for the dynamic group.

COMPARTMENT

A collection of related resources. Compartments are a fundamental component of Oracle Cloud Infrastructure for organizing and isolating your cloud resources. You use them to clearly separate resources for the purposes of measuring usage and billing, access (through the use of policies), and isolation (separating the resources for one project or business unit from another). A common approach is to create a compartment for each major part of your organization. For more information, see "Setting Up Your Tenancy" in the *Oracle Cloud Infrastructure Getting Started Guide*.

TENANCY

The root compartment that contains *all* of your organization's Oracle Cloud Infrastructure resources. Oracle automatically creates your company's tenancy for you. Directly within the tenancy are your IAM entities (users, groups, compartments, and some policies; you can also put policies into compartments inside the tenancy). You place the other types of cloud resources (e.g., instances, virtual networks, block storage volumes, etc.) inside the compartments that you create.

POLICY

A document that specifies who can access which resources, and how. Access is granted at the group and compartment level, which means you can write a policy that gives a group a specific type of access within a specific compartment, or to the tenancy itself. If you give a group access to the tenancy, the group automatically gets the same type of access to all the compartments inside the tenancy. For more information, see Example Scenario and How Policies Work. The word "policy" is used by people in different ways: to mean an individual statement written in the policy language; to mean a collection of statements in a single, named "policy" document (which has an Oracle Cloud ID (OCID) assigned to it); and to mean the overall body of policies your organization uses to control access to resources.

HOME REGION

The region where your IAM resources reside. All IAM resources are global and available across all regions, but the master set of definitions reside in a single region, the home region. You must make changes to your IAM resources in your home region. The changes will be automatically propagated to all regions. For more information, see Managing Regions.

FEDERATION

A relationship that an administrator configures between an identity provider and a service provider. When you federate Oracle Cloud Infrastructure with an identity provider, you manage users and groups in the identity provider. You manage authorization in Oracle Cloud Infrastructure's IAM service. Oracle Cloud Infrastructure tenancies are federated with Oracle Identity Cloud Service by default.

Services You Can Control Access To

You can write policies to control access to all of the services within Oracle Cloud Infrastructure.

The Administrators Group and Policy

When your company signs up for an Oracle account and Identity Domain, Oracle sets up a *default administrator* for the account. This person will be the first IAM user for your company and will be responsible for initially setting up additional administrators. Your tenancy comes with a group called *Administrators*, and the default administrator automatically belongs in this group. You can't delete this group, and there must always be at least one user in it.

Your tenancy also automatically has a policy that gives the Administrators group access to all of the Oracle Cloud Infrastructure API operations and all of the cloud resources in your tenancy. You can neither change nor delete this policy. Any other users you put into the Administrators group will have full access to all of the services. This means they can create and manage IAM resources such as, groups, policies, and compartments. And they can create and manage the cloud resources such as virtual cloud networks (VCNs), instances, block storage volumes, and any other new types of Oracle Cloud Infrastructure resources that become available in the future.

Example Scenario

The goal of this scenario is to show how the different IAM components work together, and basic features of policies.

In this scenario, Acme Company has two teams that will be using Oracle Cloud Infrastructure resources for infrastructure: Project A and Project B. In reality, your company may have many more.

Acme Company plans to use a single virtual cloud network (VCN) for both teams, and wants a network administrator to manage the VCN.

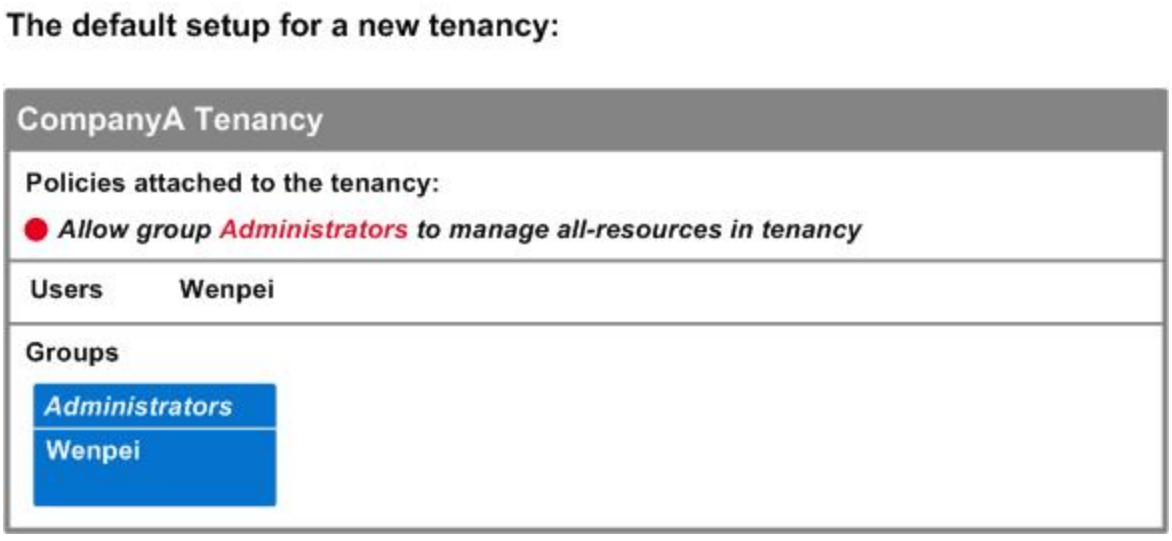
Acme Company also wants the Project A team and Project B team to *each* have their own set of instances and block storage volumes. The Project A team and Project B teams shouldn't be able to use each other's instances. These two teams also shouldn't be allowed to change anything about the VCN set up by the network administrator. Acme Company wants each team to have administrators for that team's resources. The administrators for the Project A team can decide who can use the Project A cloud resources, and how. Same for the Project B team.

Acme Company Gets Started with Oracle Cloud Infrastructure

Acme Company signs up to use Oracle Cloud Infrastructure and tells Oracle that an employee named Wenpei will be the default administrator. In response, Oracle:

- Creates a tenancy for Acme Company (see the following diagram).
- Creates an IAM user account for Wenpei in the tenancy.
- Creates the Administrators group in the tenancy and places Wenpei in that group.
- Creates a policy in Acme Company's tenancy that gives the Administrators group access to manage all of the resources in the tenancy. Here's that policy:

```
Allow group Administrators to manage all-resources in tenancy
```

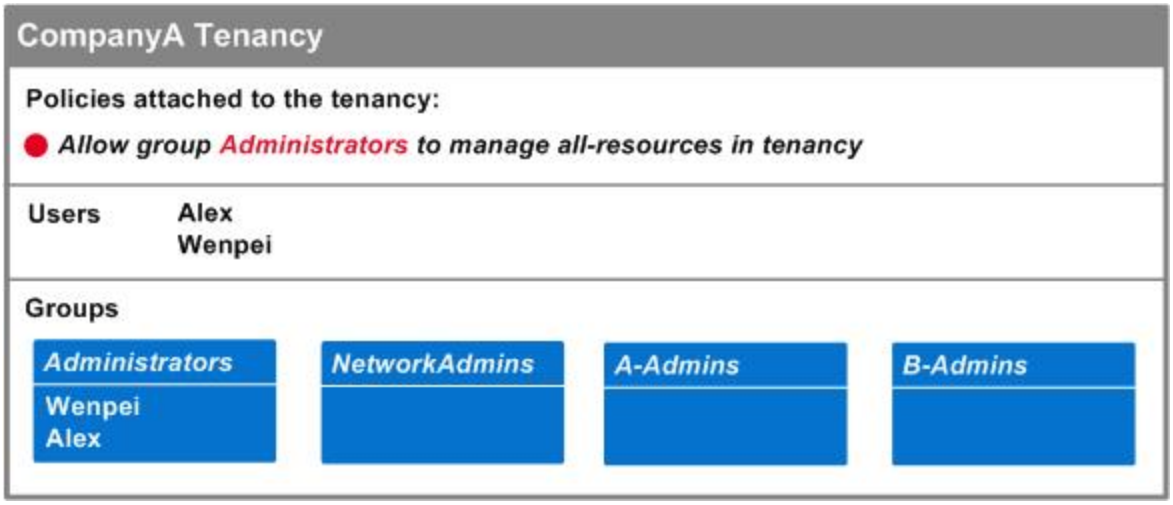


The Default Administrator Creates Some Groups and Another Administrator

Wenpei next creates several groups and users (see the following diagram). She:

- Creates groups called *NetworkAdmins*, *A-Admins*, and *B-Admins* (these last two are for Project A and Project B within the company)
- Creates a user called Alex and puts him in the Administrators group.
- Leaves the new groups empty.

To learn how to create groups, see Working with Groups. To learn how to create users and put them in groups, see Working with Users.



The Default Administrator Creates Some Compartments and Policies

Wenpei next creates compartments to group resources together (see the following diagram). She:

- Creates a compartment called *Networks* to control access to the Acme Company's VCN, subnets, IPSec VPN, and other components from Networking.
- Creates a compartment called *Project-A* to organize Project A team's cloud resources and control access to them.

- Creates a compartment called *Project-B* to organize Project B team's cloud resources and control access to them.

To learn how to manage compartments, see Working with Compartments.

Wenpei then creates a policy to give the administrators for each compartment their required level of access. She attaches the policy to the tenancy, which means that only users with access to manage policies in the tenancy can later update or delete the policy. In this scenario, that is only the Administrators group. The policy includes multiple statements that:

- Give the NetworkAdmins group access to manage networks and instances (for the purposes of easily testing the network) in the Networks compartment
- Give both the A-Admins and B-Admins groups access to use the networks in the Networks compartment (so they can create instances into the network).
- Give the A-Admins group access to manage all resources in the Project-A compartment.
- Give the B-Admins group access to manage all resources in the Project-B compartment.

Here's what that policy looks like (notice it has multiple statements in it):

```
Allow group NetworkAdmins to manage virtual-network-family in compartment Networks
Allow group NetworkAdmins to manage instance-family in compartment Networks

Allow group A-Admins,B-Admins to use virtual-network-family in compartment Networks

Allow group A-Admins to manage all-resources in compartment Project-A

Allow group B-Admins to manage all-resources in compartment Project-B
```

Notice the difference in the verbs (`manage, use`), as well as the resources (`virtual-network-family, instance-family, all-resources`). For more information about them, see Verbs and Resource-Types.To learn how to create policies, see To create a policy.



Important

A-Admins and B-Admins can *use* the virtual-network-family in the compartment Networks. However, they can't create instances in that compartment. They can only create instances in the Project-A or Project-B compartment. Remember, a compartment is a logical grouping, not a physical one, so resources that make up or reside on the same VCN can belong to different compartments.

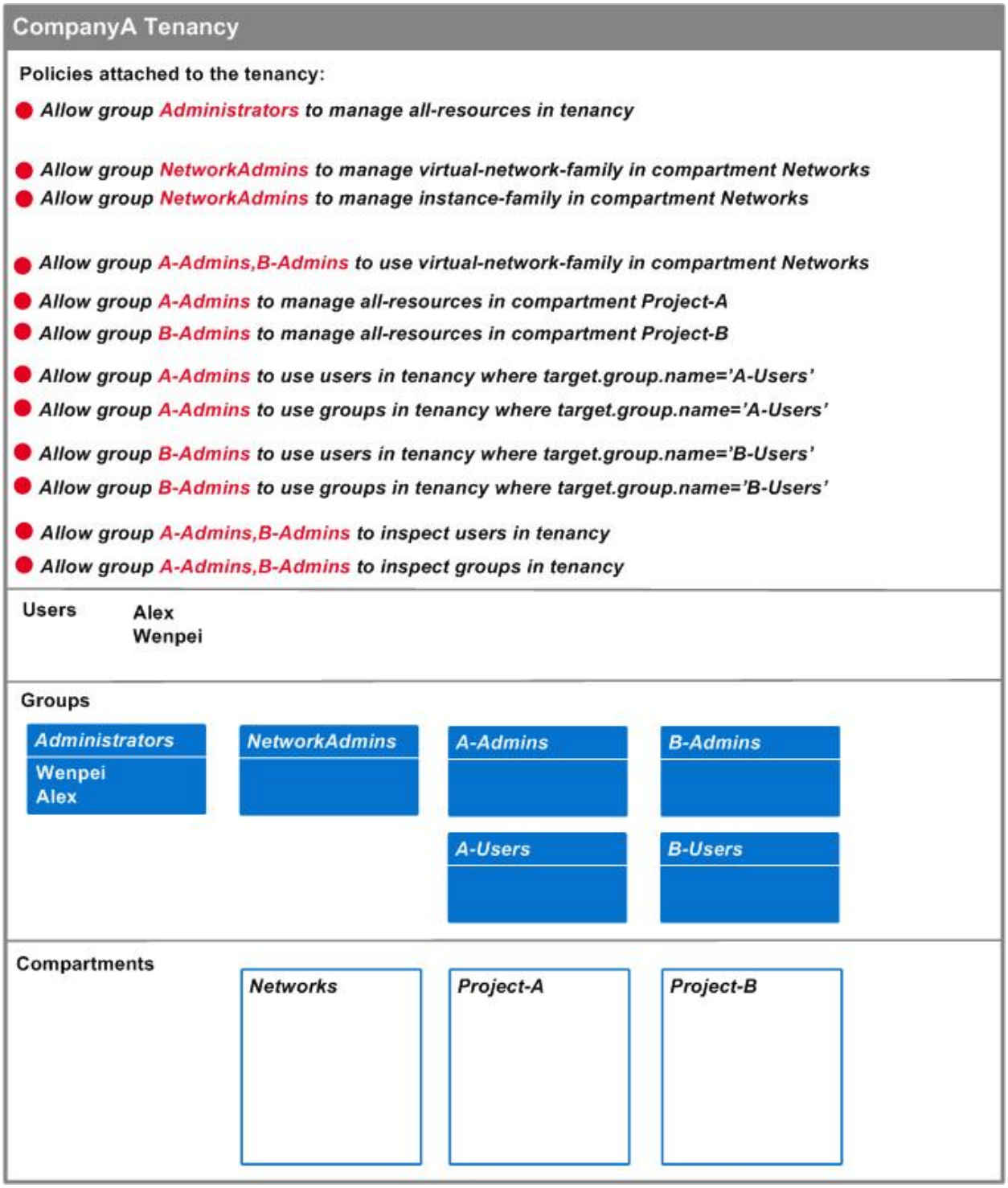
Acme Company wants to let the administrators of the Project-A and Project-B compartments decide which users can use the resources in those compartments. So Wenpei creates two more groups: A-Users and B-Users. She then adds six more statements that give the compartment admins the required access they need in order to add and remove users from those groups:

```
Allow group A-Admins to use users in tenancy where target.group.name='A-Users'
Allow group A-Admins to use groups in tenancy where target.group.name='A-Users'

Allow group B-Admins to use users in tenancy where target.group.name='B-Users'
Allow group B-Admins to use groups in tenancy where target.group.name='B-Users'

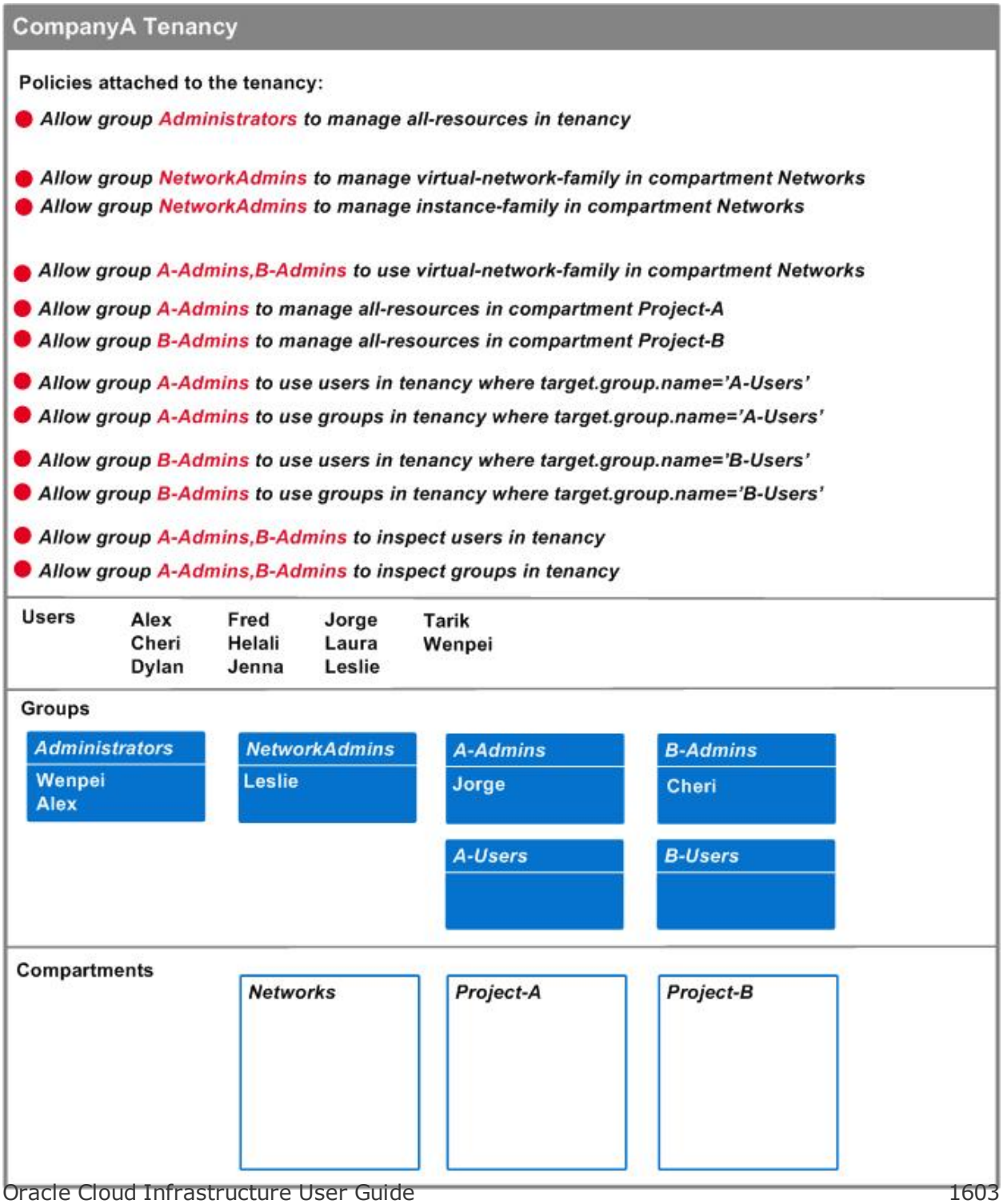
Allow group A-Admins,B-Admins to inspect users in tenancy
Allow group A-Admins,B-Admins to inspect groups in tenancy
```

Notice that this policy doesn't let the project admins *create* new users or manage credentials for the users. It lets them decide which existing users can be in the A-Users and B-Users groups. The last two statements are necessary for A-Admins and B-Admins to list all the users and groups, and confirm which users are in which groups.


CompanyA Tenancy
Policies attached to the tenancy:
Allow group Administrators to manage all-resources in tenancy
Allow group NetworkAdmins to manage virtual-network-family in compartment Networks
Allow group NetworkAdmins to manage instance-family in compartment Networks
Allow group A-Admins,B-Admins to use virtual-network-family in compartment Networks
Allow group A-Admins to manage all-resources in compartment Project-A
Allow group B-Admins to manage all-resources in compartment Project-B
Allow group A-Admins to use users in tenancy where target.group.name='A-Users'
Allow group A-Admins to use groups in tenancy where target.group.name='A-Users'
Allow group B-Admins to use users in tenancy where target.group.name='B-Users'
Allow group B-Admins to use groups in tenancy where target.group.name='B-Users'
Allow group A-Admins,B-Admins to inspect users in tenancy
Allow group A-Admins,B-Admins to inspect groups in tenancy
Users
Alex
Wenpei
Groups
Administrators
Wenpei
Alex
NetworkAdmins
A-Admins
B-Admins
A-Users
B-Users
Compartments
Networks
Project-A
Project-B

An Administrator Creates New Users

At this point, Alex is in the Administrators group and now has access to create new users. So he provisions users named Leslie, Jorge, and Cheri and places them in the NetworkAdmins, A-Admins, and B-Admins groups, respectively. Alex also creates other users who will eventually be put in the A-Users and B-Users groups by the admins for Project A and Project B.


CompanyA Tenancy
Policies attached to the tenancy:
Allow group Administrators to manage all-resources in tenancy
Allow group NetworkAdmins to manage virtual-network-family in compartment Networks
Allow group NetworkAdmins to manage instance-family in compartment Networks
Allow group A-Admins,B-Admins to use virtual-network-family in compartment Networks
Allow group A-Admins to manage all-resources in compartment Project-A
Allow group B-Admins to manage all-resources in compartment Project-B
Allow group A-Admins to use users in tenancy where target.group.name='A-Users'
Allow group A-Admins to use groups in tenancy where target.group.name='A-Users'
Allow group B-Admins to use users in tenancy where target.group.name='B-Users'
Allow group B-Admins to use groups in tenancy where target.group.name='B-Users'
Allow group A-Admins,B-Admins to inspect users in tenancy
Allow group A-Admins,B-Admins to inspect groups in tenancy
Users
Alex
Cheri
Dylan
Fred
Helali
Jenna
Jorge
Laura
Leslie
Tarik
Wenpei
Groups
Administrators
Wenpei
Alex
NetworkAdmins
Leslie
A-Admins
Jorge
B-Admins
Cheri
A-Users
B-Users
Compartments
Networks
Project-A
Project-B

The Network Admin Sets Up the Network

Leslie (in the NetworkAdmins group) has access to manage `virtual-network-family` and `instance-family` in the Networks compartment. She creates a virtual cloud network (VCN) with a single subnet in that compartment. She also sets up an Internet gateway for the VCN, and updates the VCN's route table to allow traffic via that gateway. To test the VCN's connectivity to the on-premises network, she launches an instance in the subnet in the VCN. As part of the launch request, she must specify which compartment the instance should reside in. She specifies the Networks compartment, which is the only one she has access to. She then confirms connectivity from the on-premises network to the VCN by logging in to the instance via SSH from the on-premises network.

Leslie terminates her test instance and lets Jorge and Cheri know that the VCN is up and running and ready to try out. She lets them know that their compartments are named Project-A and Project-B respectively. For more information about setting up a cloud network, see Overview of Networking. For information about launching instances into the network, see Overview of the Compute Service.

Compartment Admins Set Up Their Compartments

Jorge and Cheri now need to set up their respective compartments. Each admin needs to do the following:

- Launch instances in their own compartment
- Put users in their "users" group (e.g., A-Users)
- Decide the type of access to give those users, and accordingly attach a policy to their compartment

Jorge and Cheri both launch instances into the subnet in the VCN, into their respective team's compartments. They create and attach block volumes to the instances. Only the compartment admins can launch/terminate instances or attach/detach block olumes in their respective team's compartments.



Important

Network Topology and Compartment Access Are Different Concepts

It's important to understand the difference between the network topology of the VCN and the access control that the compartments provide. The instances Jorge launched reside in the VCN from a network topology standpoint. But from an access standpoint, they're in the Project-A compartment, not the Networks compartment where the VCN is. Leslie (the Networks admin) can't terminate or reboot Jorge's instances, or launch new ones into the Project-A compartment. But Leslie controls the instances' network, so she controls what traffic will be routed to them. If Jorge had specified the Networks compartment instead of the Project-A compartment when launching his instances, his request would have been denied. The story is similar for Cheri and the Project-B compartment.

But it's also important to note that Wenpei and Alex in the Administrators group do have access to the resources inside the compartments, because they have access to manage all kinds of resources in the tenancy. Compartments inherit any policies attached to their parent compartment (the tenancy), so the Administrators access also applies to all compartments *within* the tenancy.

Next, Jorge puts several of the users that Alex created into the A-Users group. Cheri does the same for B-Users.

Then Jorge writes a policy that gives users the level of access they need in the Project-A compartment.

```
Allow group A-Users to use instance-family in compartment Project-A
Allow group A-Users to use volume-family in compartment Project-A
Allow group A-Users to inspect virtual-network-family in compartment Networks
```

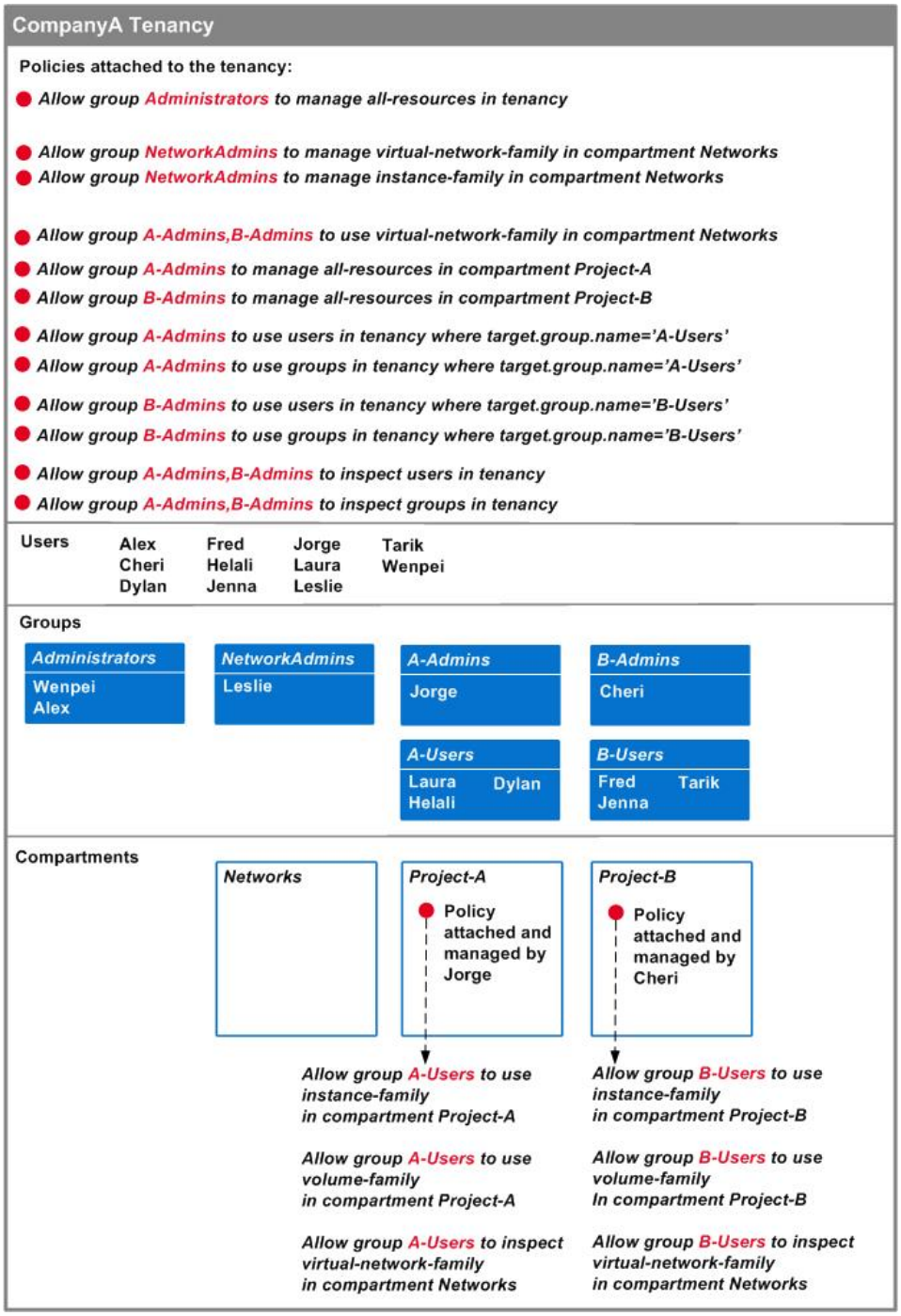
This lets them use existing instances (with attached block volumes) that the compartment admins already launched in the compartment, and stop/start/reboot them. It does not let A-Users create/delete or attach/detach any volumes. To give that ability, the policy would need to include `manage volume-family`.

Jorge attaches this policy to the Project-A compartment. Anyone with the ability to manage policies *in the compartment* can now modify or delete this policy. Right now, that is only the A-Admins group (and the Administrators group, which can do anything throughout the tenancy).

Cheri creates and attaches her own policy to the Project-B compartment, similar to Jorge's policy:

```
Allow group B-Users to use instance-family in compartment Project-B
Allow group B-Users to use volume-family in compartment Project-B
Allow group B-Users to inspect virtual-network-family in compartment Networks
```

Now the A-Users and B-Users can work with the existing instances and attached volumes in the Project-A and Project-B compartments, respectively. Here's what the layout looks like:


CompanyA Tenancy
Policies attached to the tenancy:
Allow group Administrators to manage all-resources in tenancy
Allow group NetworkAdmins to manage virtual-network-family in compartment Networks
Allow group NetworkAdmins to manage instance-family in compartment Networks
Allow group A-Admins,B-Admins to use virtual-network-family in compartment Networks
Allow group A-Admins to manage all-resources in compartment Project-A
Allow group B-Admins to manage all-resources in compartment Project-B
Allow group A-Admins to use users in tenancy where target.group.name='A-Users'
Allow group A-Admins to use groups in tenancy where target.group.name='A-Users'
Allow group B-Admins to use users in tenancy where target.group.name='B-Users'
Allow group B-Admins to use groups in tenancy where target.group.name='B-Users'
Allow group A-Admins,B-Admins to inspect users in tenancy
Allow group A-Admins,B-Admins to inspect groups in tenancy
Users
Alex
Cheri
Dylan
Fred
Helali
Jenna
Jorge
Laura
Leslie
Tarik
Wenpei
Groups
Administrators
Wenpei
Alex
NetworkAdmins
Leslie
A-Admins
Jorge
B-Admins
Cheri
A-Users
Laura
Helali
Dylan
B-Users
Fred
Jenna
Tarik
Compartments
Networks
Project-A
Policy attached and managed by Jorge
Project-B
Policy attached and managed by Cheri
Allow group A-Users to use instance-family in compartment Project-A
Allow group A-Users to use volume-family in compartment Project-A
Allow group A-Users to inspect virtual-network-family in compartment Networks
Allow group B-Users to use instance-family in compartment Project-B
Allow group B-Users to use volume-family In compartment Project-B
Allow group B-Users to inspect virtual-network-family in compartment Networks

For more information about basic and advanced features of policies, see How Policies Work. For examples of other typical policies your organization might use, see Common Policies.

Viewing Resources by Compartment in the Console

In the Console, you view your cloud resources *by compartment*. This means that after you sign in to the Console, you'll choose which compartment to work in (there's a list of the compartments you have access to on the left side of the page). Notice that compartments can be nested inside other compartments. The page will update to show that compartment's resources that are within the current region. If there are none, or if you don't have access to the resource in that compartment, you'll see a message.

This experience is different when you're viewing the lists of users, groups, dynamic groups, and federation providers. Those reside in the tenancy itself (the root compartment), not in an individual compartment.

As for policies, they can reside in either the tenancy or a compartment, depending on where the policy *is attached.* Where it's attached controls who has access to modify or delete it. For more information, see Policy Attachment.

The Scope of IAM Resources

Oracle Cloud Infrastructure uses the concepts of regions and availability domains (see Regions and Availability Domains). Some resources are available regionally, whereas others are available only within a certain availability domain. IAM resources (users, groups, dynamic groups, compartments, tag namespaces, federation providers, and policies) are global and available across all regions. See Managing Regions.

Resource Identifiers

Most types of Oracle Cloud Infrastructure resources have a unique, Oracle-assigned identifier called an Oracle Cloud ID (OCID). For information about the OCID format and other ways to identify your resources, see Resource Identifiers.

Ways to Access Oracle Cloud Infrastructure

You can access Oracle Cloud Infrastructure using the Console (a browser-based interface) or the REST API. Instructions for the Console and API are included in topics throughout this guide. For a list of available SDKs, see Software Development Kits and Command Line Interface.

To access the Console, you must use a supported browser. Oracle Cloud Infrastructure supports the latest desktop versions of Google Chrome, Microsoft Edge, Internet Explorer 11, Safari, Firefox, and Firefox ESR. Note that private browsing mode is not supported for Firefox, Internet Explorer, or Edge. Mobile browsers are not supported.

For general information about using the API, see REST APIs.

Limits on IAM Resources

See Service Limits for a list of applicable limits and instructions for requesting a limit increase.

Getting Started with Policies

If you're new to Oracle Cloud Infrastructure Identity and Access Management (IAM) policies, this topic gives guidance on how to proceed.

If You're Doing a Proof-of-Concept

If you're just trying out Oracle Cloud Infrastructure or doing a proof-of-concept project with infrastructure resources, you may not need more than a few administrators with full access to everything. In that case, you can simply create any new users you need and add them to the Administrators group. The users will be able to do anything with any kind of resource. And you can create all your resources directly in the tenancy (the root compartment). You don't need to create any compartments yet, or any other policies beyond the Tenant Admin Policy, which automatically comes with your tenancy and can't be changed.



Note

Don't forget to add your new users to the Administrators group; it's easy to forget to do that after creating them.

If You're Past the Proof-of-Concept Phase

If you're past the proof-of-concept phase and want to restrict access to your resources, first:

- Make sure you're familiar with the basic IAM components, and read through the example scenario: Overview of Oracle Cloud Infrastructure Identity and Access Management
- Think about how to organize your resources into compartments: See "Setting Up Your Tenancy" in the *Oracle Cloud Infrastructure Getting Started Guide*
- Learn the basics of how policies work: How Policies Work
- Check out some typical policies: Common Policies
- Read the FAQs below

Policy FAQs

Which of the services within Oracle Cloud Infrastructure can I control access to through policies?

All of them, including IAM itself. You can find specific details for writing policies for each service in the Policy Reference.

Can users do anything without an administrator writing a policy for them?

Yes. All users can automatically do these things without an explicit policy:

- Change or reset their own Console password.
- Manage their own API signing keys and other credentials.

Why should I separate resources by compartment? Couldn't I just put all the resources into one compartment and then use policies to control who has access to what?

You could put all your resources into a single compartment and use policies to control access, but then you would lose the benefits of measuring usage and billing by compartment, simple policy administration at the compartment level, and clear separation of resources between projects or business units.

Can I control or deny access to an individual user?

Yes. However, there are a couple things to know first:

- Enterprise companies typically have multiple users that need similar permissions, so policies are designed to give access to *groups* of users, not individual users. A user gains access by being in a group.
- Policies are designed to *allow* access; there's no explicit "deny" when you write a policy.

If you need to restrict a particular user's access, you can:

- Remove the user from the particular group of interest
- Delete the user entirely from IAM (you have to remove the user from all groups first)

How do I delete a user?

First ensure the user isn't in any groups. Only then can you delete the user.

How do I delete a compartment?

See Deleting Compartments.

How can I tell which policies apply to a particular group or user?

You need to look at the individual statements in all your policies to see which statements apply to which group. There's not currently an easy way to get this information.

How can I tell which policies apply to a particular compartment?

You need to look at the individual statements in all the policies in the tenancy to see if any apply to the particular compartment. You also need to look at any policies in the compartment itself. Policies in any of the sibling compartments *cannot* refer to the compartment of interest, so you don't need to check those policies.

How Policies Work

This topic describes how policies work and the basic features.

Overview of Policies

A *policy* is a document that specifies who can access which Oracle Cloud Infrastructure resources that your company has, and how. A policy simply allows a group to work in certain ways with specific types of resources in a particular compartment. If you're not familiar with users, groups, or compartments, see Overview of Oracle Cloud Infrastructure Identity and Access Management.

In general, here's the process an IAM administrator in your organization needs to follow:

1. Define users, groups, and one or more compartments to hold the cloud resources for your organization.
2. Create one or more policies, each written in the policy language. See Common Policies.
3. Place users into the appropriate groups depending on the compartments and resources they need to work with.
4. Provide the users with the one-time passwords that they need in order to access the Console and work with the compartments. For more information, see User Credentials.

After the administrator completes these steps, the users can access the Console, change their one-time passwords, and work with specific cloud resources as stated in the policies.

Policy Basics

To govern control of your resources, your company will have at least one policy. Each policy consists of one or more policy *statements* that follow this basic syntax:

```
Allow group <group_name> to <verb> <resource-type> in compartment <compartment_name>
```

Notice that the statements always begin with the word `Allow`. Policies only *allow* access; they cannot deny it. Instead there's an implicit deny, which means by default, users can do nothing and have to be granted access through policies. (There's one exception to this rule; see Can users do anything without an administrator writing a policy for them?)

An administrator in your organization defines the groups and compartments in your tenancy. Oracle defines the possible verbs and resource-types you can use in policies (see Verbs and Resource-Types).

In some cases you'll want the policy to apply to the tenancy and not a compartment inside the tenancy. In that case, change the end of the policy statement like so:

```
Allow group <group_name> to <verb> <resource-type> in tenancy
```

For more details about the syntax, see Policy Syntax.

For information about how many policies you can have, see Service Limits.

A Few Examples

Let's say your administrator creates a group called *HelpDesk* whose job is to manage users and their credentials. Here is a policy that enables that:

```
Allow group HelpDesk to manage users in tenancy
```

Notice that because users reside in the tenancy (the root compartment), the policy simply states the word `tenancy`, without the word `compartment` in front of it.

Next, let's say you have a compartment called *Project-A*, and a group called *A-Admins* whose job is to manage all of the Oracle Cloud Infrastructure resources in the compartment. Here's an example policy that enables that:

```
Allow group A-Admins to manage all-resources in compartment Project-A
```

Be aware that the policy directly above includes the ability to write policies *for that compartment*, which means A-Admins can control access to the compartment's resources. For more information, see Policy Attachment.

If you wanted to limit A-Admins' access to only launching and managing compute instances and block storage volumes (both the volumes and their backups) in the Project-A compartment, but the network itself lives in the Networks compartment, then the policy could instead be:

```
Allow group A-Admins to manage instance-family in compartment Project-A

Allow group A-Admins to manage volume-family in compartment Project-A

Allow group A-Admins to use virtual-network-family in compartment Networks
```

The third statement with the `virtual-network-family` resource-type enables the instance launch process, because the cloud network is involved. Specifically, the launch process creates a new VNIC and attaches it to the subnet where the instance resides.

For additional examples, see Common Policies.

Details about Specifying Groups and Compartments

Typically you'll specify a group or compartment by name in the policy. However, you can use the OCID instead. Just make sure to add "`id`" before the OCID. For example:

```
Allow group

 id ocid1.group.oc1..aaaaaaaaqjihfhvxmumrl3isyrjw3n6c4rzwskaawuc7i5xwe6s7qmnsbc6a

 to manage instance-family in compartment Project-A
```

You can specify multiple groups separated by commas:

```
Allow group A-Admins, B-Admins to manage instance-family in compartment Projects-A-and-B
```

Verbs

Oracle defines the possible verbs you can use in your policies. Here's a summary of the verbs, from least amount of access to the most:

Verb	Types of Access Covered	Target User
`inspect`	Ability to list resources, without access to any confidential information or user-specified metadata that may be part of that resource. **Important:** The operation to list policies includes the contents of the policies themselves, and the list operations for the Networking resource-types return all the information (e.g., the contents of security lists and route tables).	Third-party auditors
`read`	Includes `inspect` plus the ability to get user-specified metadata and the actual resource itself.	Internal auditors
`use`	Includes `read` plus the ability to work with existing resources (the actions vary by resource type). Includes the ability to update the resource, except for resource-types where the "update" operation has the same effective impact as the "create" operation (e.g., `UpdatePolicy`, `UpdateSecurityList`, etc.), in which case the "update" ability is available only with the `manage` verb. In general, this verb does not include the ability to create or delete that type of resource.	Day-to-day end users of resources
`manage`	Includes all permissions for the resource.	Administrators

The verb gives a certain general type of access (e.g., `inspect` lets you list and get resources). When you then join that type of access with a particular resource-type in a policy (e.g., `Allow group XYZ to inspect compartments in the tenancy`), then you give that group access to a specific set of permissions and API operations (e.g., `ListCompartments`, `GetCompartment`). For more examples, see Details for Verbs + Resource-Type Combinations.

The Policy Reference includes a similar table for each service, giving you a list of exactly which API operations are covered for each combination of verb and resource-type.

There are some special exceptions or nuances for certain resource-types.

Users: Access to both `manage users` and `manage groups` lets you do anything with users and groups, including creating and deleting users and groups, and adding/removing users from groups. To add/remove users from groups without access to creating and deleting users and groups, only both `use users` and `use groups` are required. See Common Policies.

Policies: The ability to update a policy is available only with `manage policies`, not `use policies`, because updating a policy is similar in effect to creating a new policy (you can overwrite the existing policy statements). In addition, `inspect policies` lets you get the full contents of the policies.

Object Storage objects: `inspect objects` lets you list all the objects in a bucket and do a HEAD operation for a particular object. In comparison, `read objects` lets you download the object itself.

Load Balancing resources: Be aware that `inspect load-balancers` lets you get *all* information about your load balancers and related components (backend sets, etc.).

Networking resources:

Be aware that the `inspect` verb not only returns general information about the cloud network's components (for example, the name and OCID of a security list, or of a route table). It also includes the contents of the component (for example, the actual rules in the security list, the routes in the route table, and so on).

Also, the following types of abilities are available only with the `manage` verb, not the `use` verb:

- Update (enable/disable) `internet-gateways`
- Update `security-lists`
- Update `route-tables`
- Update `dhcp-options`
- Attach a dynamic routing gateway (DRG) to a virtual cloud network (VCN)

- Create an IPSec connection between a DRG and customer-premises equipment (CPE)
- Peer VCNs



Important

Each VCN has various components that directly affect the behavior of the network (route tables, security lists, DHCP options, Internet Gateway, and so on). When you create one of these components, you establish a relationship between that component and the VCN, which means you must be allowed in a policy to both create the component and manage the VCN itself. However, the ability to *update* that component (to change the route rules, security list rules, and so on) does NOT require permission to manage the VCN itself, even though changing that component can directly affect the behavior of the network. This discrepancy is designed to give you flexibility in granting least privilege to users, and not require you to grant excessive access to the VCN just so the user can manage other components of the network. Be aware that by giving someone the ability to update a particular type of component, you're implicitly trusting them with controlling the network's behavior.

Resource-Types

Oracle also defines the resource-types you can use in your policies. First, there are *individual* types. Each individual type represents a specific type of resource. For example, the `vcns` resource-type is specifically for virtual cloud networks (VCNs).

To make policy writing easier, there are *family* types that include multiple individual resource-types that are often managed together. For example, the `virtual-network-family`

type brings together a variety of types related to the management of VCNs (e.g., `vcns`, `subnets`, `route-tables`, `security-lists`, etc.). If you need to write a more granular policy that gives access to only an individual resource-type, you can. But you can also easily write a policy to give access to a broader range of resources.

In another example: Block Volume has `volumes`, `volume-attachments`, and `volume-backups`. If you need to give access to only making backups of volumes, you can specify the `volume-backups` resource-type in your policy. But if you need to give broad access to all of the Block Volume resources, you can specify the family type called `volume-family`. For a full list of the resource-types, see Resource-Types.



Important

If a service introduces new individual resource-types, they will typically be included in the family type for that service. For example, if Networking introduces a new individual resource-type, it will be automatically included in the definition of the `virtual-network-family` resource type. For more information about future changes to the definitions of resource-types, see Policies and Service Updates.

Note that there are other ways to make policies more granular, such as the ability to specify conditions under which the access is granted. For more information, see Advanced Policy Features.

Access that Requires Multiple Resource-Types

Some API operations require access to multiple resource-types. For example, `LaunchInstance` requires the ability to create instances and work with a cloud network. The `CreateVolumeBackup` operation requires access to both the volume and the volume backup. That means you'll have separate statements to give access to each resource-type (for an example, see Let volume backup admins manage only backups). These individual statements do not have to be in the same policy. And a user can gain the required access from being in

different groups. For example, George could be in one group that gives the required level of access to the `volumes` resource-type, and in another group that gives the required access to the `volume-backups` resource-type. The sum of the individual statements, regardless of their location in the overall set of policies, gives George access to `CreateVolumeBackup`.

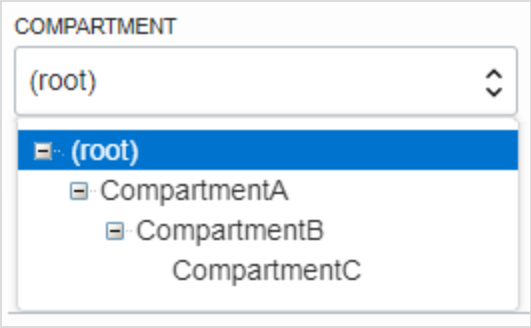
Policy Inheritance

A basic feature of policies is the concept of *inheritance*: Compartments inherit any policies from their parent compartment. The simplest example is the Administrators group, which automatically comes with your tenancy (see The Administrators Group and Policy). There's a built-in policy that enables the Administrators group to do anything in the tenancy:

```
Allow group Administrators to manage all-resources in tenancy
```

Because of policy inheritance, the Administrators group can also do anything in *any* of the compartments in the tenancy.

To illustrate further, consider a tenancy with three levels of compartments: CompartmentA, CompartmentB, and ComparmentC, shown here:



Policies that apply to resources in CompartmentA also apply to resources in CompartmentB and CompartmentC. So this policy:

```
Allow group NewtworkAdmins to manage virtual-network-family in compartment CompartmentA
```

allows the group NetworkAdmins to manage VCNs in CompartmentA, CompartmentB, and CompartmentC.

Policy Attachment

Another basic feature of policies is the concept of *attachment*. When you create a policy you must attach it to a compartment (or the tenancy, which is the root compartment). **Where you attach it controls who can then modify it or delete it.** If you attach it to the tenancy (in other words, if the policy *is in* the root compartment), then anyone with access to manage policies in the tenancy can then change or delete it. Typically that's the Administrators group or any similar group you create and give broad access to. Anyone with access only to a child compartment cannot modify or delete that policy.

If you instead attach the policy to a child compartment, then anyone with access to manage the policies *in that compartment* can change or delete it. In practical terms, this means it's easy to give compartment administrators (i.e., a group with access to `manage all-resources` in the compartment) access to manage their own compartment's policies, without giving them broader access to manage policies that reside in the tenancy. For an example that uses this kind of compartment administrator design, see Example Scenario. (Recall that because of policy inheritance, users with access to manage policies in the tenancy automatically have the ability to manage policies in compartments inside the tenancy.)

The process of attaching the policy is easy (whether attaching to a compartment or the tenancy): If you're using the Console, when you add the policy to IAM, simply make sure you're in the desired compartment when you create the policy. If you're using the API, you specify the OCID of the desired compartment (either the tenancy or other compartment) as part of the request to create the policy.

When you attach a policy to a compartment, you must be in that compartment *and* you must indicate directly in the statement which compartment it applies to. If you are not in the compartment, you'll get an error if you try to attach the policy to a different compartment. Notice that attachment occurs during policy creation, which means a policy can be attached to only one compartment. To learn how to attach a policy to a compartment, see To create a policy.

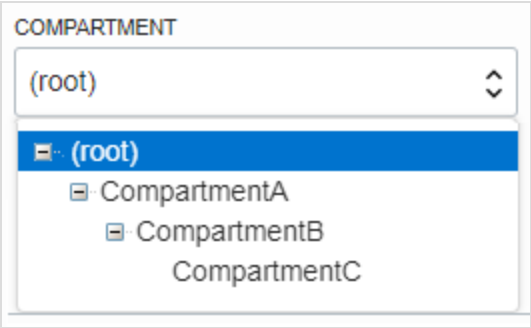
Policies and Compartment Hierarchies

As described in the previous section, a policy statement must specify the compartment for which access is being granted (or the tenancy). Where you create the policy determines who

can update the policy. If you attach the policy to the compartment or its parent, you can simply specify the compartment name. If you attach the policy further up the hierarchy, you must specify the path. The format of the path is each compartment name (or OCID) in the path, separated by a colon:

<compartment_level_1>:*<compartment_level_2>*:. . . *<compartment_level_n>*

For example, assume you have a three-level compartment hierarchy, shown here:



You want to create a policy to allow NetworkAdmins to manage VCNs in CompartmentC. If you want to attach this policy to CompartmentC or to its parent, CompartmentB, write this policy statement:

```
Allow group NewtworkAdmins to manage virtual-network-family in compartment CompartmentC
```

However, if you want to attach this policy to CompartmentA (so that only administrators of CompartmentA can modify it), write this policy statement that specifies the path:

```
Allow group NewtworkAdmins to manage virtual-network-family in compartment CompartmentB:CompartmentC
```

To attach this policy to the tenancy, write this policy statement that specifies the path from CompartmentA to CompartmentC:

```
Allow group NewtworkAdmins to manage virtual-network-family in compartment
CompartmentA:CompartmentB:CompartmentC
```

Policies and Service Updates

It's possible that the definition of a verb or resource-type could change in the future. For example, let's say that the `virtual-network-family` resource-type changes to include a new kind of resource that's been added to Networking. By default, your policies automatically stay current with any changes in service definition, so any policy you have that gives access to `virtual-network-family` would automatically include access to the newly added resource. If you'd prefer a different behavior, see Policy Language Version.

Writing Policies for Each Service

The Policy Reference includes details of the specific resource-types for each service, and which verb + resource-type combination gives access to which API operations.

Common Policies

This section includes some common policies you might want to use in your organization.



Note

These policies use example group and compartment names. Make sure to replace them with your own names.

Let the Help Desk manage users

Type of access: Ability to create, update, and delete users and their credentials. It does not include the ability to put users in groups (see Let group admins manage group membership).

Where to create the policy: In the tenancy, because users reside in the tenancy.

```
Allow group HelpDesk to manage users in tenancy
```

Let auditors inspect your resources

Type of access: Ability to list the resources in all compartments. Be aware that:

- The operation to list IAM policies includes the contents of the policies themselves
- The list operations for Networking resource-types return all the information (for example, the contents of security lists and route tables)
- The operation to list instances requires the `read` verb instead of `inspect`, and the contents include the user-provided metadata.
- The operation to view Audit service events requires the `read` verb instead of `inspect`.

Where to create the policy: In the tenancy. Because of the concept of *policy inheritance*, auditors can then inspect both the tenancy and all compartments beneath it. Or you could choose to give auditors access to only specific compartments if they don't need access to the entire tenancy.

```
Allow group Auditors to inspect all-resources in tenancy

Allow group Auditors to read instances in tenancy

Allow group Auditors to read audit-events in tenancy
```

Let network admins manage a cloud network

Type of access: Ability to manage all components in Networking. This includes cloud networks, subnets, gateways, virtual circuits, security lists, route tables, and so on. If the network admins need to launch instances to test network connectivity, see Let users launch Compute instances on this page.

Where to create the policy: In the tenancy. Because of the concept of policy inheritance, NetworkAdmins can then manage a cloud network in any compartment. To reduce the scope of access to a particular compartment, specify that compartment instead of the tenancy.

```
Allow group NetworkAdmins to manage virtual-network-family in tenancy
```

Let network admins manage load balancers

Type of access: Ability to manage all components in Load Balancing. If the group needs to launch instances, see Let users launch Compute instances on this page.

Where to create the policy: In the tenancy. Because of the concept of policy inheritance, NetworkAdmins can then manage load balancers in any compartment. To reduce the scope of access to a particular compartment, specify that compartment instead of the tenancy.

```
Allow group NetworkAdmins to manage load-balancers in tenancy
```

If the group uses the Console to manage load balancers, an additional policy to use the associated networking resources is required:

```
Allow group NetworkAdmins to manage load-balancers in tenancy

Allow group NetworkAdmins to use virtual-network-family in tenancy
```

If a particular group needs to update existing load balancers (for example, modify the backend set) but not create or delete them, use this statement:

```
Allow group LBUsers to use load-balancers in tenancy
```

Let users launch Compute instances

Type of access: Ability to do everything with instances launched into the cloud network and subnets in compartment XYZ, and attach/detach any existing volumes that already exist in compartment ABC. The first statement also lets the group create and manage instance images in compartment ABC. If the group doesn't need to attach/detach volumes, you can delete the third statement.

Where to create the policy: The easiest approach is to put this policy in the tenancy. If you want the admins of the individual compartments (ABC and XYZ) to have control over the individual policy statements for their compartments, see Policy Attachment.

```
Allow group InstanceLaunchers to manage instance-family in compartment ABC

Allow group InstanceLaunchers to read app-catalog-listing in tenancy

Allow group InstanceLaunchers to use volume-family in compartment ABC
```

```
Allow group InstanceLaunchers to use virtual-network-family in compartment XYZ
```

Let users manage Compute instance configurations and instance pools

Type of access: Ability to do all things with instance configurations and instance pools in all compartments.

Where to create the policy: In the tenancy, so that the access is easily granted to all compartments by way of policy inheritance. To reduce the scope of access to just the instance configurations and instance pools in a particular compartment, specify that compartment instead of the tenancy.

```
Allow group InstancePoolAdmins to manage compute-management-family in tenancy
```

If a particular group needs to start, stop, or reset the instances in existing instance pools, but not create or delete instance pools, use this statement:

```
Allow group InstancePoolUsers to use instance-pools in tenancy
```

Let users manage Compute autoscaling configurations

Type of access: Ability to create, update, and delete autoscaling configurations.

Where to create the policy: In the tenancy, so that the access is easily granted to all compartments by way of policy inheritance. To reduce the scope of access to just the autoscaling configurations in a particular compartment, specify that compartment instead of the tenancy.

```
Allow group AutoscalingAdmins to manage auto-scaling-configurations in tenancy

Allow group AutoscalingAdmins to manage instance-pools in tenancy
```

Let users list and subscribe to images from the Partner Image catalog

Type of access: Ability to list and create subscriptions to images in the Partner Image catalog. It does not include the ability to create instances using images from the Partner

Image catalog (see Let users launch Compute instances).

Where to create the policy: In the tenancy. To reduce the scope of access to just creating subscriptions in a particular compartment, specify that compartment instead of the tenancy in the third statement.

```
Allow group CatalogSubscribers to inspect app-catalog-listing in tenancy

Allow group CatalogSubscribers to read app-catalog-listing in tenancy

Allow group CatalogSubscribers to manage app-catalog-listing in tenancy
```

Let volume admins manage block volumes, backups, and volume groups

Type of access: Ability to do all things with block storage volumes, volume backups, and volume groups in all compartments with the exception of copying volume backups across regions. This makes sense if you want to have a single set of volume admins manage all the volumes, volume backups, and volume groups in all the compartments. The second statement is required in order to attach/detach the volumes from instances.

Where to create the policy: In the tenancy, so that the access is easily granted to all compartments by way of policy inheritance. To reduce the scope of access to just the volumes/backups and instances in a particular compartment, specify that compartment instead of the tenancy.

```
Allow group VolumeAdmins to manage volume-family in tenancy

Allow group VolumeAdmins to use instance-family in tenancy
```

If the group needs to also copy volume backups across regions, add the following statement to the policy:

```
Allow group VolumeAdmins to copy volume-backups in tenancy
```

Let volume backup admins manage only backups

Type of access: Ability to do all things with volume backups, but not create and manage

volumes themselves. This makes sense if you want to have a single set of volume backup admins manage all the volume backups in all the compartments. The first statement gives the required access to the volume that is being backed up; the second statement enables creation of the backup (and the ability to delete backups).

Where to create the policy: In the tenancy, so that the access is easily granted to all compartments by way of policy inheritance. To reduce the scope of access to just the volumes and backups in a particular compartment, specify that compartment instead of the tenancy.

```
Allow group VolumeBackupAdmins to use volumes in tenancy

Allow group VolumeBackupAdmins to manage volume-backups in tenancy
```

If the group will be using the Console, the following policy gives a better user experience:

```
Allow group VolumeBackupAdmins to use volumes in tenancy

Allow group VolumeBackupAdmins to manage volume-backups in tenancy

Allow group VolumeBackupAdmins to inspect volume-attachments in tenancy

Allow group VolumeBackupAdmins to inspect instances in tenancy

Allow group VolumeBackupAdmins to inspect backup-policies in tenancy
```

The last two statements are not necessary in order to manage volume backups. However, they enable the Console to display all the information about a particular volume and the available backup policies.

Let users create a volume group

Type of access: Ability to create a volume group from a set of volumes.

Where to create the policy: In the tenancy, so that the access is easily granted to all compartments by way of policy inheritance. To reduce the scope of access to just the volumes and volume groups in a particular compartment, specify that compartment instead of the tenancy.

```
Allow group VolumeGroupCreators to inspect volumes in tenancy
Allow group VolumeGroupCreators to manage volume-groups in tenancy
```

Let users clone a volume group

Type of access: Ability to clone a volume group from an existing volume group.

Where to create the policy: In the tenancy, so that the access is easily granted to all compartments by way of policy inheritance. To reduce the scope of access to just the volumes and volume groups in a particular compartment, specify that compartment instead of the tenancy.

```
Allow group VolumeGroupCloners to inspect volumes in tenancy
Allow group VolumeGroupCloners to manage volume-groups in tenancy
Allow group VolumeGroupCloners to manage volumes in tenancy
```

Let users create a volume group backup

Type of access: Ability to create a volume group backup.

Where to create the policy: In the tenancy, so that the access is easily granted to all compartments by way of policy inheritance. To reduce the scope of access to just the volumes/backups and volume groups/volume group backups in a particular compartment, specify that compartment instead of the tenancy.

```
Allow group VolumeGroupBackupAdmins to inspect volume-groups in tenancy
Allow group VolumeGroupBackupAdmins to manage volumes in tenancy
Allow group VolumeGroupBackupAdmins to manage volume-group-backups in tenancy
Allow group VolumeGroupBackupAdmins to manage volume-backups in tenancy
```

Let users restore a volume group backup

Type of access: Ability to create a volume group by restoring a volume group backup.

Where to create the policy: In the tenancy, so that the access is easily granted to all compartments by way of policy inheritance. To reduce the scope of access to just the volumes/backups and volume groups/volume group backups in a particular compartment, specify that compartment instead of the tenancy.

```
Allow group VolumeGroupBackupAdmins to inspect volume-group-backups in tenancy
Allow group VolumeGroupBackupAdmins to read volume-backups in tenancy
```

```
Allow group VolumeGroupBackupAdmins to manage volume-groups in tenancy
Allow group VolumeGroupBackupAdmins to manage volumes in tenancy
```

Let users create, manage, and delete file systems

Type of access: Ability to create, manage, or delete a file system. Administrative functions for a file system include the ability to rename or delete it or disconnect from it.

Where to create the policy: In the tenancy, so that the ability to create, manage, or delete a file system is easily granted to all compartments by way of policy inheritance. To reduce the scope of these administrative functions to file systems in a particular compartment, specify that compartment instead of the tenancy.

```
Allow group StorageAdmins to manage file-family in tenancy

Allow group StorageAdmins to manage file-family in compartment ABC
```

Let users create file systems

Type of access: Ability to create a file system.

Where to create the policy: In the tenancy, so that the ability to create a file system is easily granted to all compartments by way of policy inheritance. To reduce the scope of these administrative functions to file systems in a particular compartment, specify that compartment instead of the tenancy.

```
Allow group Managers to manage file-systems in tenancy

Allow group Managers to read mount-targets in tenancy
```

The second statement is required when users create a file system using the Console. It enables the Console to display a list of mount targets that the new file system can be associated with.

Let Object Storage admins manage buckets and objects

Type of access: Ability to do all things with Object Storage buckets and objects in all

compartments.

Where to create the policy: In the tenancy, so that the access is easily granted to all compartments by way of policy inheritance. To reduce the scope of access to just the buckets and objects in a particular compartment, specify that compartment instead of the tenancy.

```
Allow group ObjectAdmins to manage buckets in tenancy

Allow group ObjectAdmins to manage objects in tenancy
```

Let users write objects to Object Storage buckets

Type of access: Ability to write objects to any Object Storage bucket in compartment ABC (imagine a situation where a client needs to regularly write log files to a bucket). This consists of the ability to list the buckets in the compartment, list the objects in a bucket, and create a new object in a bucket. Although the second statement gives broad access with the `manage` verb, that access is then scoped down to only the `OBJECT_INSPECT` and `OBJECT_CREATE` permissions with the condition at the end of the statement.

Where to create the policy: The easiest approach is to put this policy in the tenancy. If you want the admins of compartment ABC to have control over the policy, see Policy Attachment.

```
Allow group ObjectWriters to read buckets in compartment ABC

Allow group ObjectWriters to manage objects in compartment ABC where any {request.permission='OBJECT_
CREATE', request.permission='OBJECT_INSPECT'}
```

Access limited to a specific bucket: To limit access to a specific bucket in a particular compartment, add the condition `where target.bucket.name='<bucket_name>'`. The following policy allows the user to list all the buckets in a particular compartment, but they can only list the objects in and upload objects to BucketA:

```
Allow group ObjectWriters to read buckets in compartment ABC

Allow group ObjectWriters to manage objects in compartment ABC where all {target.bucket.name='BucketA',
any {request.permission='OBJECT_CREATE', request.permission='OBJECT_INSPECT'}}
```

For more information about using conditions, see Advanced Policy Features.

Let users download objects from Object Storage buckets

Type of access: Ability to download objects from any Object Storage bucket in compartment ABC. This consists of the ability to list the buckets in the compartment, list the objects in a bucket, and read existing objects in a bucket.

Where to create the policy: The easiest approach is to put this policy in the tenancy. If you want the admins of compartment ABC to have control over the policy, see Policy Attachment.

```
Allow group ObjectReaders to read buckets in compartment ABC

Allow group ObjectReaders to read objects in compartment ABC
```

Access limited to a specific bucket: To limit access to a specific bucket in a particular compartment, add the condition `where target.bucket.name='<bucket_name>'`. The following policy allows the user to list all buckets in a particular compartment, but they can only read the objects in and download from BucketA:

```
Allow group ObjectReaders to read buckets in compartment ABC

Allow group ObjectReaders to read objects in compartment ABC where target.bucket.name='BucketA'
```

For more information about using conditions, see Advanced Policy Features.

Let database admins manage database systems

Type of access: Ability to do all things with the Database service in all compartments. This makes sense if you want to have a single set of database admins manage all the database systems in all the compartments.

Where to create the policy: In the tenancy, so that the access is easily granted to all compartments by way of policy inheritance. To reduce the scope of access to just the database systems in a particular compartment, specify that compartment instead of the tenancy.

```
Allow group DatabaseAdmins to manage database-family in tenancy
```

Let database admins manage Autonomous Transaction Processing databases

Deprecated. See Let database admins manage Autonomous Databases for sample Autonomous Database policies covering the Autonomous Transaction Processing workload type.

Let database admins manage Autonomous Data Warehouse databases

Deprecated. See Let database admins manage Autonomous Databases for sample Autonomous Database policies covering the Autonomous Data Warehouse workload type.

Let database admins manage Autonomous Databases

Type of access: Ability to do all things with Autonomous Database instances in all compartments. This makes sense if you want to have a single set of database admins manage all the Autonomous Database databases in all the compartments.

Where to create the policy: In the tenancy, so that the access is easily granted to all compartments by way of policy inheritance. To reduce the scope of access to just the Autonomous Databases databases in a particular compartment, specify that compartment instead of the tenancy.

```
Allow group DatabaseAdmins to manage autonomous-transaction-processing-family in tenancy
```

To reduce the scope of access to either the Autonomous Data Warehouse or Autonomous Transaction Processing workload types, use a `where` clause, as in the following examples:

```
Allow group DatabaseAdmins to manage autonomous-transaction-processing-family in tenancy where
target.workloadType = 'OLTP'
```

```
Allow group DatabaseAdmins to manage autonomous-transaction-processing-family in tenancy where
target.workloadType = 'DW'
```

Let security admins manage vaults and keys

Type of access: Ability to do all things with the Key Management service in all

compartments. This makes sense if you want to have a single set of security admins manage all the vaults and keys in all compartments.

Where to create the policy: In the tenancy, so that access is easily granted to all compartments by way of policy inheritance. To reduce the scope of access to just the keys and vaults in a particular compartment, specify that compartment instead of the tenancy.

```
Allow group SecurityAdmins to manage vaults in tenancy

Allow group SecurityAdmins to manage keys in tenancy
```

Let security admins manage all keys in a specific vault in a compartment

Type of access: Ability to do all things with keys in a specific vault in compartment ABC.

Where to create the policy: The easiest approach is to put this policy in the tenancy. If you want the admins of the individual compartment (ABC) to have control over the individual policy statements for their compartment, see Policy Attachment.

```
Allow group SecurityAdmins to manage keys in compartment ABC where target.vault.id='<vault_OCID>'
```

Let security admins use a specific key in a compartment

Type of access: Ability to list, view, and perform cryptographic operations with a specific key in a compartment.

Where to create the policy: The easiest approach is to put this policy in the tenancy. If you want the admins of the individual compartment (ABC) to have control over the individual policy statements for their compartment, see Policy Attachment.

```
Allow group SecurityAdmins to use keys in compartment ABC where target.key.id='<key_OCID>'
```

Let a user group delegate key usage in a compartment

Type of access: Ability to associate an Object Storage bucket or Block Volume volume with a specific key authorized for use in a specific compartment. With this policy, a user in the

specified group does not have permission to use the key itself. Rather, by association, the key can be used by Object Storage or Block Volume on behalf of the user to create or update an encrypted bucket or volume and to encrypt or decrypt data in the bucket or volume. This policy requires that you also have a companion policy that lets Object Storage or Block Volume use the key to perform cryptographic operations.

Where to create the policy: The easiest approach is to put this policy in the tenancy. If you want the admins of the individual compartment (ABC) to have control over the individual policy statements for their compartment, see Policy Attachment.

```
Allow group ObjectWriters, VolumeWriters to use key-delegates in compartment ABC where target.key.id =
'<key_OCID>'
```

Let Block Volume and Object Storage services encrypt and decrypt volumes and buckets

Type of access: Ability to list, view, and perform cryptographic operations with all keys in compartment ABC. Because Object Storage is a regional service, it has regional endpoints. As such, you must specify the regional service name for each region where you're using Object Storage with Key Management encryption. This policy also requires that you have a companion policy that allows a user group to use the delegated key that Object Storage or Block Volume will use.

Where to create the policy: The easiest approach is to put this policy in the tenancy. If you want the admins of the individual compartment (ABC) to have control over the individual policy statements for their compartment, see Policy Attachment.

```
Allow service blockstorage, objectstorage-<region_name> to use keys in compartment ABC where
target.key.id = '<key_OCID>'
```

For Object Storage, refer to the service in policies in the following ways, depending on your region:

- objectstorage-us-phoenix-1
- objectstorage-us-ashburn-1
- objectstorage-eu-frankfurt-1

- objectstorage-uk-london-1
- objectstorage-ap-tokyo-1

To determine the region name value of an Oracle Cloud Infrastructure region, see Regions and Availability Domains.

Let group admins manage group membership

Type of access: Ability to manage the membership of a group. Does not include the ability to create or delete users, or manage their credentials (see Let the Help Desk manage users).

The first two statements let GroupAdmins list all the users and groups in the tenancy, list which users are in a particular group, and list what groups a particular user is in.

The last two statements together let GroupAdmins change a group's membership. The condition at the end of the last two statements lets GroupAdmins manage membership to all groups except the Administrators group (see The Administrators Group and Policy). You should protect membership to that group because it has power to do anything throughout the tenancy.

It might seem that the last two statements should also cover the basic listing functionality that the first two statements enable. To better understand how conditions work and why you also need the first two statements, see Variables that Aren't Applicable to a Request Result in a Declined Request.

Where to create the policy: In the tenancy, because users and groups reside in the tenancy.

```
Allow group GroupAdmins to inspect users in tenancy

Allow group GroupAdmins to inspect groups in tenancy

Allow group GroupAdmins to use users in tenancy where target.group.name != 'Administrators'

Allow group GroupAdmins to use groups in tenancy where target.group.name != 'Administrators'
```

Let users manage their own passwords and credentials

No policy is required to let users manage *their own* credentials. All users have the ability to change and reset their own passwords, manage their own API keys, and manage their own auth tokens. For more information, see User Credentials.

Let a compartment admin manage the compartment

Type of access: Ability to manage all aspects of a particular compartment. For example, a group called A-Admins could manage all aspects of a compartment called Project-A, including writing additional policies that affect the compartment. For more information, see Policy Attachment. For an example of this kind of setup and additional policies that are useful, see Example Scenario.

Where to create the policy: In the tenancy.

```
Allow group A-Admins to manage all-resources in compartment Project-A
```

Restrict admin access to a specific region

Type of access: Ability to manage resources in a specific region. Remember that IAM resources must be managed in the home region. If the specified region is not the home region, then the Admin will not be able to manage IAM resources. For more information about the home region, see Managing Regions.

Where to create the policy: In the tenancy.

```
Allow group PHX-Admins to manage all-resources in tenancy where request.region='phx'
```

The preceding policy allows PHX-Admins to manage all aspects of all resources in us-phoenix-1.

Members of the PHX-Admins group can only manage IAM resources if the tenancy's home region is us-phoenix-1.

Restrict user access to view only summary announcements

Type of access: Ability to view the summary versions of announcements about the operational status of Oracle Cloud Infrastructure services.

Where to create the policy: In the tenancy.

```
Allow group AnnouncementListers to inspect announcements in tenancy
```

The preceding policy allows AnnouncementListers to view a list of summary announcements.

Let users view details of announcements

Type of access: Ability to view the details of announcements about the operational status of Oracle Cloud Infrastructure services.

Where to create the policy: In the tenancy.

```
Allow group AnnouncementReaders to read announcements in tenancy
```

The preceding policy allows AnnouncementReaders to view a list of summary announcements and the details of specific announcements.

Let streaming users manage streams

Type of access: Ability to do all things with the Streaming service in all compartments.

Where to create the policy: In the tenancy, so that the access is easily granted to all compartments by way of policy inheritance. To reduce the scope of access to just the streams in a particular compartment, specify that compartment instead of the tenancy.

```
Allow group StreamAdmins to manage streams in tenancy
```

Let streaming users publish messages to streams

Type of access: Ability to produce messages to streams with the Streaming service in all compartments.

Where to create the policy: In the tenancy, so that the access is easily granted to all compartments by way of policy inheritance. To reduce the scope of access to just the streams in a particular compartment, specify that compartment instead of the tenancy.

```
Allow group StreamUsers to use stream-push in tenancy
```

Let streaming users publish messages to a specific stream

Type of access: Ability to produce messages to a stream with the Streaming service.

Where to create the policy: In the tenancy, so that the access is easily granted to all compartments by way of policy inheritance. To reduce the scope of access to just the streams in a particular compartment, specify that compartment instead of the tenancy.

```
allow group StreamUsers to use stream-pull in tenancy where target.stream.id = '<stream_OCID>'
```

Let streaming users consume messages from streams

Type of access: Ability to consume messages from streams with the Streaming service in all compartments.

Where to create the policy: In the tenancy, so that the access is easily granted to all compartments by way of policy inheritance. To reduce the scope of access to just the streams in a particular compartment, specify that compartment instead of the tenancy.

```
Allow group StreamUsers to use stream-pull in tenancy
```

Let users view metric definitions in a compartment

Type of access: Ability to view metric definitions in a specific compartment. For more information about metrics, see Metrics Feature Overview.

Where to create the policy: In the tenancy, so that the access is easily granted to all compartments by way of policy inheritance. To reduce the scope of access to just the metric definitions in a particular compartment, specify that compartment instead of the tenancy.

```
Allow group MetricReaders to inspect metrics in compartment ABC
```

Let users access monitoring metrics in a compartment

Type of access: Ability to view and retrieve monitoring metrics for supported resources in a specific compartment. For more information about metrics, see Metrics Feature Overview.

Where to create the policy: In the tenancy, so that the access is easily granted to all compartments by way of policy inheritance. To reduce the scope of access to just the metrics in a particular compartment, specify that compartment instead of the tenancy.

```
Allow group MetricReaders to read metrics in compartment ABC
```

Restrict user access to a specific metric namespace

Type of access: Ability to view and retrieve monitoring metrics for resources under a specific metric namespace. For more information about metrics, see Metrics Feature Overview.

Where to create the policy: In the tenancy, so that the access is easily granted to all compartments by way of policy inheritance. To reduce the scope of access to the specified metric namespace to just within a particular compartment, specify that compartment instead of the tenancy.

```
Allow group MetricReaders to read metrics in compartment ABC where target.metrics.namespace='oci_
computeagent'
```

The preceding policy allows `MetricReaders` to view and retrieve metric data points from all monitoring-enabledCompute instances in the `ABC` compartment.

Let users publish custom metrics

Type of access: Ability to publish custom metrics under a specific metric namespace to the Monitoring service. For instructions, see Publishing Custom Metrics.

Where to create the policy: In the tenancy, so that the access is easily granted to all compartments by way of policy inheritance. To reduce the scope of access to just metrics in a particular compartment, specify that compartment instead of the tenancy.

```
Allow group MetricPublishers to use metrics in tenancy where
target.metrics.namespace='mycustomnamespace'
```

The preceding policy allows `MetricPublishers` to publish data points for the custom metric namespace `mycustomnamespace` in the tenancy.

Let instances make API calls to access monitoring metrics in the tenancy

Type of access: Ability to call the Monitoring API for access to monitoring metrics. The instances on which API requests originate must be members of the dynamic group indicated in the policy. For more information, see Calling Services from an Instance and Metrics Feature Overview.

Where to create the policy: In the tenancy.

```
Allow dynamic-group MetricInstances to read metrics in tenancy
```

The preceding policy allows applications that are running on Compute instances in the dynamic group `MetricInstances` to send API requests to the Monitoring service in the tenancy.

Let users view alarms

Type of access: Ability to view alarms for supported resources in tenancy. Does not include the ability to create alarms or to create or delete topics. For more information about alarms, see Alarms Feature Overview.

Where to create the policy: In the tenancy. Because of the concept of policy inheritance, AlarmUsers can then view alarms in any compartment. To reduce the scope of access to a particular compartment, specify that compartment instead of the tenancy.

```
Allow group AlarmUsers to read alarms in tenancy

Allow group AlarmUsers to read metrics in tenancy
```

Let users manage alarms

Type of access: Ability to view and create alarms with existing notification topics for supported resources in the tenancy. Does not include the ability to create or delete topics. For more information about alarms, see Alarms Feature Overview.

All statements are required to let AlarmUsers create alarms.

Where to create the policy: In the tenancy. Because of the concept of policy inheritance, AlarmUsers can then view and create alarms in any compartment. To reduce the scope of access to a particular compartment, specify that compartment instead of the tenancy.

```
Allow group AlarmUsers to manage alarms in tenancy

Allow group AlarmUsers to read metrics in tenancy

Allow group AlarmUsers to use ons-topics in tenancy
```

Let users manage alarms and create topics

Type of access: Ability to view and create alarms (with new or existing topics) for supported resources in tenancy. For more information about alarms, see Alarms Feature Overview.

Where to create the policy: In the tenancy. Because of the concept of policy inheritance, AlarmUsers can then view and create alarms in any compartment. To reduce the scope of access to a particular compartment, specify that compartment instead of the tenancy.

```
Allow group AlarmUsers to manage alarms in tenancy

Allow group AlarmUsers to read metrics in tenancy

Allow group AlarmUsers to manage ons-topics in tenancy
```

Allow a group to manage topics

Type of access: Ability to get, create, update, and delete topics in the tenancy.

Where to create the policy: In the tenancy.

```
Allow group A-Admins to manage ons-topics in tenancy
```

Allow a group to manage topic subscriptions

Type of access: Ability to list, create, update, and delete subscriptions for topics in the tenancy.

Where to create the policy: In the tenancy.

```
Allow group A-Admins to manage ons-subscriptions in tenancy
```

Allow a group to publish messages to topics

Type of access: Ability to broadcast notification messages to all subscriptions in the tenancy, as well as list, create, update, and delete subscriptions in the tenancy.

Where to create the policy: In the tenancy.

```
Allow group A-Admins to use ons-topics in tenancy
```

Advanced Policy Features

This section describes policy language features that let you grant more granular access.

Conditions

As part of a policy statement, you can specify one or more *conditions* that must be met in order for access to be granted. For a simple example, see Let group admins manage group membership.

Each condition consists of one or more predefined variables that you specify values for in the policy statement. Later, when someone requests access to the resource in question, if the condition in the policy is met, it evaluates to *true* and the request is allowed. If the condition is not met, it evaluates to *false* and the request is not allowed.

There are two types of variables: those that are relevant to the request itself, and those relevant to the resource(s) being acted upon in the request, also known as the *target*. The name of the variable is prefixed accordingly with either `request` or `target` followed by a period. For example, there's a request variable called `request.operation` to represent the

API operation being requested. This variable lets you write a broad policy statement, but add a condition based on the specific API operation. For an example, see Let users write objects to Object Storage buckets.

Variables that Aren't Applicable to a Request Result in a Declined Request

If the variable is *not applicable* to the incoming request, the condition evaluates to *false* and the request is declined. For example, here are the basic policy statements that together let someone add or remove users from any group except Administrators:

```
Allow group GroupAdmins to use users in tenancy
 where target.group.name != 'Administrators'

Allow group GroupAdmins to use groups in tenancy
 where target.group.name != 'Administrators'
```

Given the above policy, if GroupAdmins tried to call a general API operation for users such as `ListUsers` or `UpdateUser` (which lets you change the user's description), the request would be declined, even though those API operations are covered by `use users`.This is because the above policy statement for `use users` also includes the `target.group.name` variable, but the `ListUsers` or `UpdateUser` request doesn't involve specifying a group. There is no `target.group.name` for those requests, so the request is declined.

If you want to also grant access to general user API operations that don't involve a particular group, you would need an additional statement that gives the level of access you want to grant, *but without the condition*. For example, if you want to grant access to `ListUsers`, you need this additional statement:

```
Allow group GroupAdmins to inspect users in tenancy
```

Or if you want to grant access to `UpdateUser`, you need this additional statement (which also covers `ListUsers` because the `use` verb includes the capabilities of the `inspect` verb):

```
Allow group GroupAdmins to use users in tenancy
```

This general concept also applies to groups (e.g., `ListGroups` and `UpdateGroup`), and any other resource type with target variables.

For more information about the syntax of conditions, see Conditions. For a list of all the variables you can use in policies, see the tables in the Policy Reference.

Permissions

Permissions are the atomic units of authorization that control a user's ability to perform operations on resources. Oracle defines all the permissions in the policy language. When you write a policy giving a group access to a particular verb and resource-type, you're actually giving that group access to one or more predefined permissions. The purposes of verbs is to simplify the process of granting multiple related permissions that cover a broad set of access or a particular operational scenario. The next sections give more details and examples.

Relation to Verbs

To understand the relationship between permissions and verbs, let's look at an example. A policy statement that allows a group to `inspect volumes` actually gives the group access to a permission called VOLUME_INSPECT (permissions are always written with all capital letters and underscores). In general, that permission enables the user to get information about block volumes.

As you go from `inspect` > `read` > `use` > `manage`, the level of access generally increases, and the permissions granted are cumulative. The following table shows the permissions included with each verb for the `volumes` resource-type. Notice that no additional permissions are granted going from `inspect` to `read`.

Inspect Volumes	Read Volumes	Use Volumes	Manage Volumes
VOLUME_INSPECT	VOLUME_INSPECT	VOLUME_INSPECT VOLUME_UPDATE VOLUME_WRITE	VOLUME_INSPECT VOLUME_UPDATE VOLUME_WRITE VOLUME_CREATE VOLUME_DELETE

The policy reference lists the permissions covered by each verb for each given resource-type. For example, for block volumes and other resources covered by the Core Services, see the tables in Details for Verb + Resource-Type Combinations. The left column of each of those

tables lists the permissions covered by each verb. The other sections of the policy reference include the same kind of information for the other services.

Relation to API Operations

Each API operation requires the caller to have access to one or more permissions. For example, to use either `ListVolumes` or `GetVolume`, you must have access to a single permission: VOLUME_INSPECT. To attach a volume to an instance, you must have access to multiple permissions, some of which are related to the `volumes` resource-type, some to the `volume-attachments` resource-type, and some related to the `instances` resource-type:

- VOLUME_WRITE
- VOLUME_ATTACHMENT_CREATE
- INSTANCE_ATTACH_VOLUME

The policy reference lists which permissions are required for each API operation. For example, for the Core Services API operations, see the table in Permissions Required for Each API Operation.

Understanding a User's Access

The policy language is designed to let you write simple statements involving only verbs and resource-types, without having to state the desired permissions in the statement. However, there may be situations where a security team member or auditor wants to understand the specific permissions a particular user has. The tables in the policy reference show each verb and the associated permissions. You can look at the groups the user is in and the policies applicable to those groups, and from there compile a list of the permissions granted. However, having a list of the permissions isn't the complete picture. Conditions in a policy statement can scope a user's access beyond individual permissions (see the next section). Also, each policy statement specifies a particular compartment and can have conditions that further scope the access to only certain resources in that compartment.

Scoping Access with Permissions or API Operations

In a policy statement, you can use conditions combined with permissions or API operations to reduce the scope of access granted by a particular verb.

For example, let's say you want group XYZ to be able to list, get, create, or update groups (i.e., change their description), but not delete them. To list, get, create, and update groups, you need a policy with `manage groups` as the verb and resource-type. According to the table in Details for Verbs + Resource-Type Combinations, the permissions covered are:

- GROUP_INSPECT
- GROUP_UPDATE
- GROUP_CREATE
- GROUP_DELETE

To restrict access to only the desired permissions, you could add a condition *that explicitly states the permissions you want to allow*:

```
Allow group XYZ to manage groups in tenancy

 where any {request.permission='GROUP_INSPECT',
            request.permission='GROUP_CREATE',
            request.permission='GROUP_UPDATE'}
```

An alternative would be a policy that *allows all permissions except* GROUP_DELETE:

```
Allow group XYZ to manage groups in tenancy where request.permission != 'GROUP_DELETE'
```

However, with this approach, be aware that any new permissions the service might add in the future would automatically be granted to group XYZ. Only GROUP_DELETE would be omitted.

Another alternative would be to write a condition *based on the specific API operations*. Notice that according to the table in Permissions Required for Each API Operation, both `ListGroups` and `GetGroup` require only the GROUP_INSPECT permission. Here's the policy:

```
Allow group XYZ to manage groups in tenancy

 where any {request.operation='ListGroups',
            request.operation='GetGroup',
            request.operation='CreateGroup',
            request.operation='UpdateGroup'}
```

It can be beneficial to use permissions instead of API operations in conditions. In the future, if a new API operation is added that requires one of the permissions listed in the permissions-based policy above, that policy will already control XYZ group's access to that new API operation.

But notice that you can further scope a user's access to a permission by *also* specifying a condition based on API operation. For example, you could give a user access to GROUP_INSPECT, but then only to `ListGroups`.

```
Allow group XYZ to manage groups in tenancy

 where all {request.permission='GROUP_INSPECT',
            request.operation='ListGroups'}
```

Policy Language Version

As mentioned in Policies and Service Updates, by default your policies stay current with any changes to the definitions of verbs and resources as the services change. If you'd prefer to limit access according to the definitions that were current on a specific date, you can do that. When creating a policy, you can specify a date, and that is considered its "version". You can also update the version for an existing policy. For more information, see To create a policy and also To update the version date for an existing policy.

Policy Syntax

The overall syntax of a policy statement is as follows:

```
Allow <subject> to <verb> <resource-type> in <location> where <conditions>
```

Spare spaces or line breaks in the statement have no effect.

For limits on the number of policies and statements, see Service Limits.

Subject

Specify one or more comma-separated groups by name or OCID. Or specify `any-user` to cover all users in the tenancy.

Syntax: `group <group_name> | group id <group_ocid> | dynamic-group <dynamic-group_name> | dynamic-group id<dynamic-group_ocid> | any-user`

Examples:

- To specify a single group by name:

```
Allow group A-Admins to manage all-resources in compartment Project-A
```

- To specify multiple groups by name (a space after the comma is optional):

```
Allow group A-Admins, B-Admins to manage all-resources in compartment Projects-A-and-B
```

- To specify a single group by OCID (the OCID is shortened for brevity):

```
Allow group

 id ocid1.group.oc1..aaaaaaaaqjihfhvxmum...awuc7i5xwe6s7qmnsbc6a

to manage all-resources in compartment Project-A
```

- To specify multiple groups by OCID (the OCIDs are shortened for brevity):

```
Allow group

 id ocid1.group.oc1..aaaaaaaaqjihfhvxmumrl...wuc7i5xwe6s7qmnsbc6a,
 id ocid1.group.oc1..aaaaaaaavhea5mellwzb...66yfxvl462tdgx2oecyq

to manage all-resources in compartment Projects-A-and-B
```

- To specify any user in the tenancy:

```
Allow any-user to inspect users in tenancy
```

Verb

Specify a single verb. For a list of verbs, see Verbs. Example:

```
Allow group A-Admins to manage all-resources in compartment Project-A
```

Resource-Type

Specify a single resource-type, which can be one of the following:

- An individual resource-type (e.g., `vcns`, `subnets`, `instances`, `volumes`, etc.)
- A family resource-type (e.g., `virtual-network-family`, `instance-family`, `volume-family`, etc.)
- `all-resources`: Covers all resources in the compartment (or tenancy).

A family resource-type covers a variety of components that are typically used together. This makes it easier to write a policy that gives someone access to work with various aspects of your cloud network.

For a list of the available resource-types, see Resource-Types.

Syntax: *<resource_type>* | all-resources

Examples:

- To specify a single resource-type:

```
Allow group HelpDesk to manage users in tenancy
```

- To specify multiple resource-types, use separate statements:

```
Allow group A-Users to manage instance-family in compartment Project-A

Allow group A-Users to manage volume-family in compartment Project-A
```

- To specify all resources in the compartment (or tenancy):

```
Allow group A-Admins to manage all-resources in compartment Project-A
```

Location

Specify a single compartment or compartment path by name or OCID. Or simply specify `tenancy` to cover the entire tenancy. Remember that users, groups, and compartments reside in the tenancy. Policies can reside in (i.e., be attached to) either the tenancy or a child compartment.



Note

Granting Access to Specific Regions or Availability Domains

To create a policy that gives access to a specific region or availability domain, use the `request.region` or `request.ad` variable with a condition. See Conditions.

The location is required in the statement. If you want to attach a policy to a compartment, you must be in that compartment when you create the policy. For more information, see Policy Attachment.

To specify a compartment that is not a direct child of the compartment you are attaching the policy to, specify the path to the compartment, using the colon (:) as a separator. For more information, see Policies and Compartment Hierarchies.

Syntax: `[ tenancy | compartment <compartment_name> | compartment id <compartment_ocid> ]`

Examples:

- To specify a compartment by name:

```
Allow group A-Admins to manage all-resources in compartment Project-A
```

- To specify a compartment by OCID:

```
Allow group
 id ocid1.group.oc1..aaaaaaaavhea5mellwzbmplwrpum46xfc73sb4rm66yfxvl462tdgx2oecyq

to manage all-resources in compartment
 id ocid1.compartment.oc1..aaaaaaaayzfq...4fmameqh7lcdlihrvur7xq
```

- To specify multiple compartments, use separate statements:

```
Allow group InstanceAdmins to manage instance-family in compartment Project-A

Allow group InstanceAdmins to manage instance-family in compartment Project-B
```

- To specify multiple compartments by OCID, use separate statements:

```
Allow group id
 ocd1.group.oc1..aaaaaaaavhea5mell...b4rm66yfxvl462tdgx2oecyq
to manage all-resources in compartment id
 ocid1.compartment.oc1..aaaaaaaayzfqei...ameq4h7lcdlihrvur7xq


Allow group id
 ocd1.group.oc1..aaaaaaaavhea5mell...b4rm66yfxvl462tdgx2oecyq
to manage all-resources in compartment id
 ocid1.compartment.oc1..aaaaaaaaphfjutov5s...vyypllbtctehnqg756a
```

- To specify a compartment that is not a direct child of the compartment where you are attaching the policy, specify the path:

```
Allow group InstanceAdmins to manage instance-family in compartment Project-A:Project-A2
```

Conditions

Specify one or more conditions. Use `any` or `all` with multiple conditions for a logical OR or AND, respectively.

Syntax for a single condition: `variable =|!= value`

Syntax for multiple conditions: `any|all {<condition>,<condition>,...}`

For a list of variables supported by all the services, see General Variables for All Requests. Also see the details for each service in the Policy Reference. Here are the types of values you can use in conditions:

Type	Examples
String	`'johnsmith@example.com'` `'ocid1.compartment.oc1..aaaaaaaaph...ctehnqg756a'` (single quotation marks are required around the value)
Pattern	`/HR*/` (matches strings that start with "HR") `/*HR/` (matches strings that end with "HR") `/*HR*/` (matches strings that contain "HR")

Examples:

- A single condition.
  The following policy enables the GroupAdmins group to create, update, or delete any groups with names that start with "A-Users-":

  ```
  Allow group GroupAdmins to manage groups in tenancy where target.group.name = /A-Users-*/
  ```

  The following policy enables the GroupAdmins group to manage the membership of any group besides the Administrators group:

  ```
  Allow group GroupAdmins to use users in tenancy where target.group.name != 'Administrators'

  Allow group GroupAdmins to use groups in tenancy where target.group.name != 'Administrators'
  ```

  The following policy enables the NetworkAdmins group to manage cloud networks in any compartment except the one specified:

  ```
  Allow group NetworkAdmins to manage virtual-network-family in tenancy where target.compartment.id
  != 'ocid1.compartment.oc1..aaaaaaaayzfqeibduyox6icmdol6zyar3ugly4fmameq4h7lcdlihrvur7xq'
  ```

- Multiple conditions.
  The following policy lets A-Admins create, update, or delete any groups whose names start with "A-", except for the A-Admins group itself:

  ```
  Allow group GroupAdmins to manage groups in tenancy where

   all {target.group.name=/A-*/,target.group.name!='A-Admins'}
  ```

Note that in the above policies, the statements do not let GroupAdmins actually list all the users and groups. To understand why not, see Variables that Aren't Applicable to a Request Result in a Declined Request.

Policy Reference

This reference includes:

- Verbs: A list of the available actions to pair with a resource-type
- Resource-Types: A list of the main resource-types
- General Variables for All Requests: Variables you can use when writing policies for any resource-type
- Details for the Announcements Service
- Details for the Audit Service
- Details for Container Engine for Kubernetes
- Details for the Core Services (this includes Networking, Compute, and Block Volume)
- Details for the Database Service
- Details for the DNS Service
- Details for the Email Service
- Details for the File Storage Service
- Details for the Health Checks Service
- Details for IAM
- Details for the Key Management Service
- Details for Load Balancing
- Details for Monitoring
- Details for the Notifications Service
- Details for Object Storage, Archive Storage, and Data Transfer
- Details for Registry

- Details for Resource Manager
- Details for the Search Service
- Details for the Streaming Service
- Details for the WAF Service

For instructions on how to create and manage policies using the Console or API, see Managing Policies.

Verbs

The verbs are listed in order of least amount of ability to most. The exact meaning of a each verb depends on which resource-type it's paired with. The tables later in this section show the API operations covered by each combination of verb and resource-type.

Verb	Types of Access Covered	Target User
`inspect`	Ability to list resources, without access to any confidential information or user-specified metadata that may be part of that resource. **Important:** The operation to list policies includes the contents of the policies themselves, and the list operations for the Networking resource-types return all the information (e.g., the contents of security lists and route tables).	Third-party auditors
`read`	Includes `inspect` plus the ability to get user-specified metadata and the actual resource itself.	Internal auditors

Verb	Types of Access Covered	Target User
`use`	Includes `read` plus the ability to work with existing resources (the actions vary by resource type). Includes the ability to update the resource, except for resource-types where the "update" operation has the same effective impact as the "create" operation (e.g., `UpdatePolicy`, `UpdateSecurityList`, etc.), in which case the "update" ability is available only with the `manage` verb. In general, this verb does not include the ability to create or delete that type of resource.	Day-to-day end users of resources
`manage`	Includes all permissions for the resource.	Administrators

Resource-Types

The family resource-types are listed below. For the individual resource-types that make up each family, follow the links.

- `all-resources`: All Oracle Cloud Infrastructure resource-types
- `cluster-family`: See Details for Container Engine for Kubernetes
- `database-family`: See Details for the Database Service
- `dns`: See Details for the DNS Service
- `file-family`: See Details for the File Storage Service
- `instance-family`: See Details for the Core Services
- `object-family`: See Details for Object Storage, Archive Storage, and Data Transfer
- `virtual-network-family`: See Details for the Core Services
- `volume-family`: See Details for the Core Services

IAM has no family resource-type, only individual ones. See Details for IAM.

General Variables for All Requests

You use variables when adding conditions to a policy. For more information, see Conditions. Here are the general variables applicable to all requests.

Name	Type	Description
`request.user.id`	Entity (OCID)	The OCID of the requesting user.
`request.groups.id`	List of entities (OCIDs)	The OCIDs of the groups the requesting user is in.
`target.compartment.id`	Entity (OCID)	The OCID of the compartment containing the primary resource. **Note:** `target.compartment.id` and `target.compartment.name` cannot be used with a "List" API operation to filter the list based on the requesting user's access to the compartment.
`target.compartment.name`	String	The name of the compartment specified in `target.compartment.id`.
`request.operation`	String	The API operation name being requested (e.g., ListUsers).
`request.permission`	String	The underlying permission(s) being requested (see Permissions).

Name	Type	Description
`request.region`	String	The key of the region the request is made in. Allowed values are: • `fra` • `iad` • `icn` • `lhr` • `nrt` • `phx` • `yyz`
`request.ad`	String	The name of the availability domain the request is made in. To get a list of availability domain names, use the ListAvailabilityDomains operation.

Details for the Announcements Service

This topic covers details for writing policies to control access to the Announcements service.

Resource-Types

- `announcements`

Supported Variables

Only the general variables are supported (see General Variables for All Requests).

Details for Verb + Resource-Type Combinations

The following tables show the permissions and API operations covered by each verb. The level of access is cumulative as you go from `inspect` > `read` > `use` > `manage`. A plus sign (+) in a table cell indicates incremental access compared to the cell directly above it, whereas "no extra" indicates no incremental access.

For example, the `read` verb for the `announcements` resource-type includes the same permissions and API operations as the `inspect` verb, plus the ANNOUNCEMENT_READ permission and an additional API operation, `GetAnnouncement`. However, the `use` verb and `manage` verbs cover no extra permissions or API operations compared to `read`.

announcements

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
ANNOUNCEMENT_LIST	`ListAnnouncements`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT + ANNOUNCEMENT_READ	*INSPECT +* `GetAnnouncement`	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

Permissions Required for Each API Operation

The following table lists the API operations in a logical order, grouped by resource type.

For information about permissions, see Permissions.

API Operation	Permissions Required to Use the Operation
`ListAnnouncements`	ANNOUNCEMENT_LIST
`GetAnnouncement`	ANNOUNCEMENT_READ

Details for the Audit Service

This topic covers details for writing policies to control access to the Audit service.

Resource-Types

```
audit-events
```

Supported Variables

Only the general variables are supported (see General Variables for All Requests).

Details for Verb + Resource-Type Combinations

The following tables show the permissions and API operations covered by each verb. The level of access is cumulative as you go from `inspect` > `read` > `use` > `manage`. A plus sign (+) in a table cell indicates incremental access compared to the cell directly above it, whereas "no extra" indicates no incremental access.

For example, the `use` and `manage` verbs for the `audit-events` resource-type cover no extra permissions or API operations compared to the `read` verb.

AUDIT-EVENTS

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
none	*none*	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
AUDIT_EVENT_READ	`ListAuditEvents`	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

Permissions Required for Each API Operation

The following table lists the API operations in a logical order, grouped by resource type.

For information about permissions, see Permissions.

API Operation	Permissions Required to Use the Operation
`ListAuditEvents`	AUDIT_EVENT_READ

Details for the Core Services

This topic covers details for writing policies to control access to the Core Services (Networking, Compute, and Block Volume).

Resource-Types

Networking

AGGREGATE RESOURCE-TYPE

`virtual-network-family`

INDIVIDUAL RESOURCE-TYPES

`vcns`

`subnets`

`route-tables`

`security-lists`

`dhcp-options`

`private-ips`

`public-ips`

`internet-gateways`

`nat-gateways`

`service-gateways`

`local-peering-gateways` (which includes `local-peering-from`, and `local-peering-to`)

`remote-peering-connections` (which includes `remote-peering-from`, and `remote-peering-to`)

`drgs`

`drg-attachments`

`cpes`

`ipsec-connections`

`cross-connects`

`cross-connect-groups`

`virtual-circuits`

`vnics`

`vnic-attachments`

COMMENTS

A policy that uses `<verb> virtual-network-family` is equivalent to writing one with a separate `<verb> <individual resource-type>` statement for each of the individual resource-types.

See the table in Details for Verb + Resource-Type Combinations for a detailed breakout of the API operations covered by each verb, for each individual resource-type included in `virtual-network-family`.

Compute

INSTANCE-FAMILY AGGREGATE RESOURCE-TYPE

The `instance-family` aggregate resource-type covers these individual resource-types:

`app-catalog-listing`

`console-histories`

`instances`

`instance-console-connection`

`instance-images`

`volume-attachments` (includes only the permissions required for attaching volumes to instances)

COMPUTE-MANAGEMENT-FAMILY AGGREGATE RESOURCE-TYPE

The `compute-management-family` aggregate resource-type covers these individual resource-types:

`instance-configurations`

`instance-pools`

ADDITIONAL INDIVIDUAL RESOURCE-TYPES

```
auto-scaling-configurations
```

COMMENTS

A policy that uses `<verb> instance-family` or `<verb> compute-management-family` is equivalent to writing one with a separate `<verb> <individual resource-type>` statement for each of the individual resource-types in the family.

See the table in Details for Verb + Resource-Type Combinations for a detailed breakout of the API operations covered by each verb, for each individual resource-type.

Block Volume

AGGREGATE RESOURCE-TYPE

```
volume-family
```

INDIVIDUAL RESOURCE-TYPES

```
volumes
volume-attachments
volume-backups
boot-volume-backups
backup-policies
volume-groups
volume-group-backups
```

COMMENTS

A policy that uses `<verb> volume-family` is equivalent to writing one with a separate `<verb> <individual resource-type>` statement for each of the individual resource-types.

See the table in Details for Verb + Resource-Type Combinations for a detailed breakout of the API operations covered by each verb, for each individual resource-type included in `volume-family`.

Supported Variables

The Core Services support all the general variables, plus the ones listed here. For more information about general variables supported by Oracle Cloud Infrastructure services, see General Variables for All Requests.

Variable	Variable Type	Comments
`target.boot-volume.kms-key.id`	String	Use this variable to control whether Compute instances can be launched with boot volumes that were created without a Key Management master encryption key.

Details for Verb + Resource-Type Combinations

The following tables show the permissions and API operations covered by each verb. The level of access is cumulative as you go from `inspect` > `read` > `use` > `manage`. A plus sign (+) in a table cell indicates incremental access compared to the cell directly above it, whereas "no extra" indicates no incremental access.

For example, the `read` and `use` verbs for the `vcns` resource-type cover no extra permissions or API operations compared to the `inspect` verb. However, the `manage` verb includes several extra permissions and API operations.

FOR VIRTUAL-NETWORK-FAMILY RESOURCE TYPES

vcns

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
VCN_READ	`ListVcns` `GetVcn`	`CreateNatGateway, DeleteNatGateway` (both also need `manage nat-gateways` and `manage vcns`) **Note:** The above operations in this cell are totally covered with just `manage virtual-network-family`.

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*no extra*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*no extra*

MANAGE

Permissions	APIs Fully Covered
USE +	*USE* +
VCN_ATTACH	`CreateVcn`
VCN_DETACH	`UpdateVcn`
VCN_UPDATE	`DeleteVcn`
VCN_CREATE	
VCN_DELETE	

subnets

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
SUBNET_READ	`ListSubnets` `GetSubnet`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + SUBNET_ATTACH SUBNET_DETACH	*no extra*	`LaunchInstance` (also need `manage instance-family`) `TerminateInstance` (also need `manage instance-family`, and `use volumes` if a volume is attached) `AttachVnic` (also need `manage instances` and either `use vnics` or `use instance-family`) `DetachVnic` (also need `manage instances` and either `use vnics` or `use instance-family`) `CreatePrivateIp, DeletePrivateIp` (both also need `use private-ips` and `use vnics`)

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + SUBNET_CREATE SUBNET_UPDATE SUBNET_DELETE	*no extra*	*USE* + `CreateSubnet, DeleteSubnet` (both also need `manage vcns, manage route-tables, manage security-lists, manage dhcp-options`) `UpdateSubnet` (also need `manage route-tables` if changing which route table is associated with the subnet, `manage security-lists` if changing which security lists are associated with the subnet, and `manage dhcp-options` if changing which set of DHCP options is associated with the subnet) **Note:** The above operations in this cell are covered with just `manage virtual-network-family`.

route-tables

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
ROUTE_TABLE_READ	`ListRouteTables` `GetRouteTable`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + ROUTE_TABLE_ATTACH ROUTE_TABLE_DETACH ROUTE_TABLE_UPDATE ROUTE_TABLE_CREATE ROUTE_TABLE_DELETE	*no extra*	`CreateRouteTable`, `DeleteRouteTable` (both also need `manage vcns`, `manage internet-gateways`, `manage drgs`, `manage private-ips`, `manage local-peering-gateways`, `use nat-gateways`, `use service-gateways`) `UpdateRouteTable` (also need `manage internet-gateways`, `manage drgs`, `manage private-ips`, `manage local-peering-gateways`, `use nat-gateways`, `use service-gateways`) `CreateSubnet`, `DeleteSubnet` (both also need `manage vcns`, `manage subnets`, `manage security-lists`, `manage dhcp-options`) `UpdateSubnet` (if changing which route table is associated with the subnet, also need `manage subnets`) **Note:** All of the above operations in this cell are totally covered with just `manage virtual-network-family`.

security-lists

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
SECURITY_LIST_READ	`ListSecurityLists` `GetSecurityList`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + SECURITY_LIST_ATTACH SECURITY_LIST_DETACH SECURITY_LIST_UPDATE SECURITY_LIST_CREATE SECURITY_LIST_DELETE	*USE* + `UpdateSecurityList` **Note:** Ability to update a security list is available only with the `manage` verb, not the `use` verb.	`CreateSecurityList`, `DeleteSecurityList` (both also need `manage vcns`) `CreateSubnet`, `DeleteSubnet` (both also need `manage vcns`, `manage subnets`, `manage route-tables`, `manage dhcp-options`) `UpdateSubnet` (if changing which security lists are associated with the subnet, also need `manage subnets`) **Note:** All of the above operations in this cell are totally covered with just `manage virtual-network-family`.

dhcp-options

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
DHCP_READ	`ListDhcpOptions` `GetDhcpOptions`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + DHCP_ATTACH DHCP_DETACH DHCP_UPDATE DHCP_CREATE DHCP_DELETE	*USE* + `UpdateDhcpOptions` **Note:** Ability to update a set of DHCP options is available only with the `manage` verb, not the `use` verb.	*USE* + `CreateDhcpOptions, DeleteDhcpOptions` (both also need `manage vcns`) `CreateSubnet, DeleteSubnet` (also need `manage vcns, manage subnets, manage route-tables, manage security-lists`) `UpdateSubnet` (if changing which set of DHCP options is associated with the subnet, also need `manage subnets`) **Note:** All of the above operations in this cell are totally covered with just `manage virtual-network-family`.

private-ips

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
PRIVATE_IP_READ	`ListPrivateIps` `GetPrivateIp` For ephemeral public IPs only: `ListPublicIps`, `GetPublicIpByPrivateIpId`, `GetPublicIpByIpAddress`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + PRIVATE_IP_UPDATE PRIVATE_IP_ASSIGN PRIVATE_IP_UNASSIGN PRIVATE_IP_CREATE PRIVATE_IP_DELETE PRIVATE_IP_ASSIGN_PUBLIC_IP PRIVATE_IP_UNASSIGN_PUBLIC_IP	*READ* + `UpdatePrivateIp` For ephemeral public IPs: `UpdatePublicIp`, `CreatePublicIp`, `DeletePublicIp`	`CreatePrivateIp`, `DeletePrivateIp` (both also need `use subnets` and `use vnics`) For reserved public IPs: `UpdatePublicIp`, `CreatePublicIp`, `DeletePublicIp` (all also need `manage public-ips`) **Note:** The above operations in this cell are totally covered with just `use virtual-network-family`.

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + PRIVATE_IP_ROUTE_TABLE_ATTACH PRIVATE_IP_ROUTE_TABLE_DETACH	*no extra*	*USE* + `CreateRouteTable, DeleteRouteTable` (both also need `manage vcns, manage internet-gateways, manage drgs, and manage route-tables, manage local-peering-gateways, use nat-gateways, use service-gateways`) `UpdateRouteTable` (also need `manage internet-gateways, manage drgs, manage route-tables, manage local-peering-gateways, use nat-gateways, use service-gateways`) **Note:** The above operations in this cell are totally covered with just `manage virtual-network-family`.

public-ips

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
none	*none*	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
PUBLIC_IP_READ	For reserved public IPs only: `ListPublicIps`, `GetPublicIpByPrivateIpId`, `GetPublicIpByIpAddress` Permissions for listing/getting ephemeral public IPs are part of the private-ip permissions.	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + PUBLIC_IP_ASSIGN_PRIVATE_IP PUBLIC_IP_UNASSIGN_PRIVATE_IP	*no extra*	For ephemeral private IPs: `CreatePrivateIp, DeletePrivateIp` (both also need `use private-ips`) **Note:** The above operations in this cell are totally covered with just `use virtual-network-family`.

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + PUBLIC_IP_UPDATE PUBLIC_IP_CREATE PUBLIC_IP_DELETE	*no extra*	*USE* + For reserved public IPs: `UpdatePrivateIp`, `CreatePrivateIp, DeletePrivateIp` (all of these also need `use private-ips`) **Note:** The above operations in this cell are totally covered with just `manage virtual-network-family`.

internet-gateways

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
INTERNET_GATEWAY_READ	`ListInternetGateways` `GetInternetGateway`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + INTERNET_GATEWAY_ATTACH INTERNET_GATEWAY_DETACH INTERNET_GATEWAY_UPDATE INTERNET_GATEWAY_CREATE INTERNET_GATEWAY_DELETE	*USE* + `UpdateInternetGateway` **Note:** Ability to update a an internet gateway is available only with the `manage` verb, not the `use` verb.	`CreateInternetGateway`, `DeleteInternetGateway` (both also need `manage vcns`) `CreateRouteTable`, `DeleteRouteTable` (both also need `manage route-tables`, `manage vcns`, `manage drgs`, `manage private-ips`, `manage local-peering-gateways`, `use nat-gateways`, `use nat-gateways`, `use service-gateways`) `UpdateRouteTable` (also need `manage route-tables`, `manage drgs`, `manage private-ips`, `manage local-peering-gateways`, `use nat-gateways`, `use service-gateways`) **Note:** All of the above operations in this cell are totally covered with just `manage virtual-network-family`.

nat-gateways

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
none	*none*	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
NAT_GATEWAY_READ	`ListNatGateways` `GetNatGateway`	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + NAT_GATEWAY_ATTACH NAT_GATEWAY_DETACH	*no extra*	*READ* + `CreateRouteTable, DeleteRouteTable` (both also need `manage route-tables, manage vcns, manage drgs, manage private-ips, manage internet-gateways, manage local-peering-gateways, use service-gateways`) `UpdateRouteTable` (also need `manage route-tables, manage drgs, manage private-ips, manage internet-gateways, manage local-peering-gateways, use service-gateways`) **Note:** All of the above operations in this cell are totally covered with just `manage virtual-network-family`.

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + NAT_GATEWAY_UPDATE NAT_GATEWAY_CREATE NAT_GATEWAY_DELETE	*USE* + `UpdateNatGateway` **Note:** Ability to update a NAT gateway is available only with the `manage` verb, not the `use` verb.	`CreateNatGateway, DeleteNatGateway` (both also need `manage vcns`) **Note:** All of the above operations in this cell are totally covered with just `manage virtual-network-family`.

service-gateways

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
SERVICE_GATEWAY_READ	`ListServiceGateways` `GetServiceGateway`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*no extra*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + SERVICE_GATEWAY_ATTACH SERVICE_GATEWAY_DETACH	*no extra*	*READ* + `CreateRouteTable, DeleteRouteTable` (both also need `manage route-tables, manage vcns, manage internet-gateways, manage drgs, manage private-ips, manage local-peering-gateways`) `UpdateRouteTable` (also need `manage route-tables, manage drgs, manage internet-gateways, manage private-ips, manage local-peering-gateways`)

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + SERVICE_GATEWAY_UPDATE SERVICE_GATEWAY_CREATE SERVICE_GATEWAY_DELETE SERVICE_GATEWAY_ADD_SERVICE SERVICE_GATEWAY_DELETE_SERVICE	*USE* + `UpdateServiceGateway` `AttachServiceId` `DetachServiceId` **Note:** Ability to update a service gateway is available only with the `manage` verb, not the `use` verb.	`CreateServiceGateway`, `DeleteServiceGateway` (both also need `manage vcns`) **Note:** All of the above operations in this cell are totally covered with just `manage virtual-network-family`.

local-peering-gateways

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
LOCAL_PEERING_GATEWAY_READ	`ListLocalPeeringGateways` `GetLocalPeeringGateway`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	*no extra*	`CreateLocalPeeringGateway` (also need `manage vcns`, and need `manage route-tables` if you associate a route table during creation)
LOCAL_PEERING_GATEWAY_UPDATE		`UpdateLocalPeeringGateway` (also need `manage route-tables` if you associate a route table during the update)
LOCAL_PEERING_GATEWAY_ATTACH		`DeleteLocalPeeringGateway` (also need `manage vcns`)
LOCAL_PEERING_GATEWAY_DETACH		`CreateRouteTable, DeleteRouteTable` (both also need `manage route-tables, manage vcns, manage internet-gateways, manage drgs, manage private-ips, use nat-gateways, use service-gateways`)
LOCAL_PEERING_GATEWAY_CREATE		`UpdateRouteTable` (also need `manage route-tables, manage internet-gateways, manage drgs, manage private-ips, use nat-gateways, use service-gateways`)
LOCAL_PEERING_GATEWAY_DELETE		**Note:** The above operations in this cell are totally covered with just `manage virtual-network-family`.

local-peering-from

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
LOCAL_PEERING_GATEWAY_READ	*none*	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*none*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*none*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + LOCAL_PEERING_GATEWAY_CONNECT_FROM	*no extra*	`ConnectLocalPeeringGateways` (acceptor in the peering relationship must also grant the requestor `manage local-peering-to` in the compartment where the acceptor's LPG resides. See Local VCN Peering (Within Region).) **Note:** The above operation in this cell is totally covered with just `manage virtual-network-family`.

local-peering-to

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
LOCAL_PEERING_GATEWAY_READ	*none*	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*none*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*none*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + LOCAL_PEERING_GATEWAY_CONNECT_TO	*no extra*	`ConnectLocalPeeringGateways` (requestor in the peering relationship must also have `manage local-peering-from` in the compartment where the requestor's LPG resides. See Local VCN Peering (Within Region).) **Note:** The above operation in this cell is totally covered with just `manage virtual-network-family`.

remote-peering-connections

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
REMOTE_PEERING_CONNECTION_READ	`ListRemotePeeringConnections` `GetRemotePeeringConnection`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + REMOTE_PEERING_CONNECTION_UPDATE REMOTE_PEERING_CONNECTION_CREATE REMOTE_PEERING_CONNECTION_DELETE	`UpdateRemotePeeringConnection`	`CreateRemotePeeringConnection`, `DeleteRemotePeeringConnection` (both also need `manage drgs`) **Note:** The above operations in this cell are totally covered with just `manage virtual-network-family`.

remote-peering-from

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
REMOTE_PEERING_CONNECTION_READ	*none*	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*none*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*none*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + REMOTE_PEERING_CONNECTION_CONNECT_FROM	*no extra*	`ConnectRemotePeeringConnections` (acceptor in the peering relationship must also grant the requestor `manage remote-peering-to` in the compartment where the acceptor's RPC resides. See Remote VCN Peering (Across Regions).) **Note:** The above operation in this cell is totally covered with just `manage virtual-network-family`.

remote-peering-to

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
REMOTE_PEERING_CONNECTION_READ	*none*	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*none*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*none*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + REMOTE_PEERING_CONNECTION_CONNECT_TO	*no extra*	`ConnectRemotePeeringConnections` (requestor in the peering relationship must also have `manage remote-peering-from` in the compartment where the requestor's RPC resides. See Remote VCN Peering (Across Regions).) **Note:** The above operation in this cell is totally covered with just `manage virtual-network-family`.

drgs

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
DRG_READ DRG_ATTACHMENT_READ	`ListDrgs` `GetDrg` `ListDrgAttachments`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered
USE +	*USE* +
DRG_ATTACH	`CreateDrg`
DRG_DETACH	`UpdateDrg`
DRG_UPDATE	`DeleteDrg`
DRG_ATTACHMENT_UPDATE	
DRG_CREATE	
DRG_DELETE	

cpes

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
CPE_READ	`ListCpes` `GetCpe`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + CPE_ATTACH CPE_DETACH CPE_UPDATE CPE_CREATE CPE_DELETE	*USE* + `CreateCpe` `UpdateCpe` `DeleteCpe`	`CreateIPSecConnection`, `DeleteIPSecConnection` (both also need `manage ipsec-connections` and `manage drgs`) **Note:** All of the above operations in this cell are totally covered with just `manage virtual-network-family`.

ipsec

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
IPSEC_CONNECTION_READ	`ListIPSecConnections` `GetIPSecConnection` `GetIPSecConnectionStatus` `ListIPSecConnectionTunnels` `GetIPSecConnectionTunnel`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT + IPSEC_CONNECTION_DEVICE_CONFIG_READ	*INSPECT* + `GetIPSecConnectionDeviceConfig` `GetIPSecConnectionTunnelSharedSecret`	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + IPSEC_CONNECTION_CREATE IPSEC_CONNECTION_UPDATE IPSEC_CONNECTION_DELETE IPSEC_CONNECTION_DEVICE_CONFIG_ UPDATE	*USE* + `UpdateIPSecConnection` `UpdateIPSecConnectionTunnel`	`CreateIPSecConnection`, `DeleteIPSecConnection` (both also need `manage cpes` and `manage drgs`) **Note:** All of the above operations in this cell are totally covered with just `manage virtual-network-family`.

cross-connects

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
CROSS_CONNECT_READ	`ListCrossConnects` `GetCrossConnect`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*no extra*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + CROSS_CONNECT_UPDATE CROSS_CONNECT_CREATE CROSS_CONNECT_DELETE CROSS_CONNECT_ATTACH CROSS_CONNECT_DETACH	`UpdateCrossConnect` `CreateCrossConnect` `DeleteCrossConnect`	`UpdateVirtualCircuit` (also need `use virtual-circuits`) `CreateVirtualCircuit`, `DeleteVirtualCircuit` (also need `manage virtual-circuits`)

cross-connect-groups

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
CROSS_CONNECT_GROUP_READ	`ListCrossConnectGroups` `GetCrossConnectGroup`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*no extra*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + CROSS_CONNECT_GROUP_UPDATE CROSS_CONNECT_GROUP_CREATE CROSS_CONNECT_GROUP_DELETE	`UpdateCrossConnectGroup` `CreateCrossConnectGroup` `DeleteCrossConnectGroup`	*no extra*

virtual-circuits

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
VIRTUAL_CIRCUIT_READ	`ListVirtualCircuits` `GetVirtualCircuit`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + VIRTUAL_CIRCUIT_UPDATE	*no extra*	`UpdateVirtualCircuit` (also need `manage drgs`,and if you're also changing which cross-connect or cross-connect group the virtual circuit uses, also need `manage cross-connects`)

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + VIRTUAL_CIRCUIT_CREATE VIRTUAL_CIRCUIT_DELETE	*no extra*	*USE* + `CreateVirtualCircuit`, `DeleteVirtualCircuit` (both also need `manage drgs`, and if you're also creating/deleting the virtual circuit with a mapping to a specific cross-connect or cross-connect group, also need `manage cross-connects`) **Note:** All of the above operations in this cell are totally covered with just `manage virtual-network-family`.

vnics

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
VNIC_READ	`GetVnic`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + VNIC_ATTACH VNIC_DETACH VNIC_CREATE VNIC_DELETE VNIC_UPDATE	`UpdateVnic`	`LaunchInstance` (also need `use subnets` and `manage instance-family`) `AttachVnic` (also need `manage instances` and `use subnets`) `DetachVnic` (also need `manage instances` and `use subnets`) `CreatePrivateIp, DeletePrivateIp` (both also need `use subnets` and `use private-ips`)

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*no extra*

vnic-attachments

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
VNIC_ATTACHMENT_READ	`GetVnicAttachment`	`ListVnicAttachments` (also need `inspect instances`)

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*none*	*no extra*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*none*	*no extra*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*none*	*no extra*

FOR INSTANCE-FAMILY RESOURCE TYPES

The `instance-family` aggregate resource-type includes extra permissions beyond the sum of the permissions for the individual resource-types included in `instance-family`. For example: It includes a few permissions for `vnics` and `volumes`, even though those resource-types aren't generally considered part of the `instance-family`. Why are there extras included? So you can write fewer policy statements to cover general use cases, like working with an instance that has an attached block volume. You can write one statement for `instance-family` instead of multiple statements covering `instances`, `vnics`, and `volumes`.

Here's a list of the extra permissions:

For `inspect instance-family`:

- VNIC_READ
- VNIC_ATTACHMENT_READ
- VOLUME_ATTACHMENT_INSPECT

For `read instance-family`:

- VOLUME_ATTACHMENT_READ

For `use instance-family`:

- VNIC_ATTACH
- VNIC_DETACH
- VOLUME_ATTACHMENT_UPDATE

For `manage instance-family`:

- VOLUME_ATTACHMENT_CREATE
- VOLUME_ATTACHMENT_DELETE

The following tables list the permissions and API operations covered by each of the individual resource-types included in `instance-family`.

instances

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
INSTANCE_INSPECT	*none*	`GetConsoleHistory`, `ListConsoleHistories` (both also need `inspect console-histories`) `ListVnicAttachments` (also need `inspect vnic-attachments`) `ListVolumeAttachments` (also need `inspect volumes` and `inspect volume-attachments`) `GetVolumeAttachments` (also need `inspect volumes` and `inspect volume-attachments`)

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT + INSTANCE_READ	`ListInstances` `GetInstance` **Note:** `ListInstances` and `GetInstance` include any user-provided metadata added to the instance	*INSPECT* + `CaptureConsoleHistory` (also need `manage console-histories` and `read instance-images`) `ShowConsoleHistoryData` (also need `read console-histories` and `read instance-images`)

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ +	*READ* +	*READ* +
INSTANCE_UPDATE	`UpdateInstance`	`CreateImage` (also need `manage instance-images`)
INSTANCE_CREATE_IMAGE	`InstanceAction`	`AttachVolume` (also need `manage volume-attachments` and `use volumes`)
INSTANCE_POWER_ACTIONS		`DetachVolume` (also need `manage volume-attachments` and `use volumes`)
INSTANCE_ATTACH_VOLUME		
INSTANCE_DETACH_VOLUME		

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	*no extra*	*USE* +
INSTANCE_CREATE		`LaunchInstance` (also need `read instance-images`, `use vnics`, `use subnets`, and `read app-catalog-listing`)
INSTANCE_DELETE		`TerminateInstance` (also need `use vnics` and `use subnets`; also need `manage volume-attachments` and `use volumes` if a volume is attached)
INSTANCE_ATTACH_SECONDARY_VNIC		`AttachVnic` (also need `use subnets` and either `use vnics` or use `instance-family`)
INSTANCE_DETACH_SECONDARY_VNIC		`DetachVnic` (also need `use subnets` and either `use vnics` or use `instance-family`)

console-histories

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
CONSOLE_HISTORY_INSPECT	*none*	`ListConsoleHistories`, `GetConsoleHistory` (both also need `inspect instances`)

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT + CONSOLE_HISTORY_READ	*none*	*INSPECT* + `ShowConsoleHistoryData` (also need `read instances` and `read instance-images`)

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*none*	*no extra*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + CONSOLE_HISTORY_CREATE CONSOLE_HISTORY_DELETE	`DeleteConsoleHistory`	*USE* + `CaptureConsoleHistory` (also need `read instances` and `read instance-images`)

instance-console-connection

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
INSTANCE_CONSOLE_CONNECTION_INSPECT	*none*	`ListInstanceConsoleConnections` (also need `inspect instances` and `read instances`)

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT + INSTANCE_CONSOLE_CONNECTION_READ	*none*	*INSPECT* + `GetInstanceConsoleConnection` (also need `read instances`)

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ +	*none*	*no extra*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	`DeleteInstanceConsoleConnection`	`CreateInstanceConsoleConnection` (also need `read instances`)
INSTANCE_CONSOLE_CONNECTION_CREATE		
INSTANCE_CONSOLE_CONNECTION_DELETE		

instance-images

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
INSTANCE_IMAGE_INSPECT	`ListImages`	*none*
	`GetImage`	

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT +	*no extra*	*INSPECT* +
INSTANCE_IMAGE_READ		`LaunchInstance` (also need `manage instances`, `use vnics`, and `use subnets`)
		`CaptureConsoleHistory` (also need `read instances` and `manage console-histories`)
		`ShowConsoleHistoryData` (also need `read instances` and `read console-histories`)

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ +	`UpdateImage`	*no extra*
INSTANCE_IMAGE_UPDATE		

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	`DeleteImage`	*USE* +
INSTANCE_IMAGE_CREATE		`CreateImage` (also need `use instances`)
INSTANCE_IMAGE_DELETE		

app-catalog-listing

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
APP_CATALOG_LISTING_INSPECT	`ListAppCatalogListings`	*none*
	`GetAppCatalogListing`	
	`ListAppCatalogListingResourceVersions`	
	`GetAppCatalogListingResourceVersion`	
	`GetAppCatalogListingAgreements`	
	`ListAppCatalogSubscriptions`	

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT +	*none*	*INSPECT* +
APP_CATALOG_LISTING_READ		`LaunchInstance` (Also need `use instances`, `read instance-images`, `use vnics`, and `use subnets`.)

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
READ +	`CreateAppCatalogSubscription`	*none*
APP_CATALOG_LISTING_SUBSCRIBE	`DeleteAppCatalogSubscription`	

FOR COMPUTE-MANAGEMENT-FAMILY RESOURCE TYPES

The following tables list the permissions and API operations covered by each of the individual resource-types included in `compute-management-family`.

instance-configurations

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
INSTANCE_CONFIGURATION_INSPECT	`ListInstanceConfigurations`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT +	*INSPECT* +	*none*
INSTANCE_CONFIGURATION_READ	`GetInstanceConfiguration`	

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	*USE* +	*none*
INSTANCE_CONFIGURATION_CREATE	`CreateInstanceConfiguration`	
INSTANCE_CONFIGURATION_UPDATE	`UpdateInstanceConfiguration`	
INSTANCE_CONFIGURATION_LAUNCH	`LaunchInstanceConfiguration`	
INSTANCE_CONFIGURATION_DELETE	`DeleteInstanceConfiguration`	

instance-pools

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
INSTANCE_POOL_INSPECT	`ListInstancePools`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT +	*INSPECT* +	`ListInstancePoolInstances`
INSTANCE_POOL_READ	`GetInstancePool`	

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + INSTANCE_POOL_POWER_ACTIONS	*no extra*	*READ* + `ResetInstancePool` `SoftresetInstancePool` `StartInstancePool` `StopInstancePool` All also need `use instances`.

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + INSTANCE_POOL_CREATE INSTANCE_POOL_UPDATE INSTANCE_POOL_DELETE	*USE* + `UpdateInstancePool`	*USE* + `CreateInstancePool` (also need `manage instances`, `read instance-images`, `use vnics`, and `use subnets`) `TerminateInstancePool` (also need `manage instances`, `use vnics`, `use subnets`, `manage volume-attachments`, and `use volumes`)

FOR ADDITIONAL COMPUTE INDIVIDUAL RESOURCE TYPES

The following table lists the permissions and API operations covered by other Compute resource-types that aren't included in any aggregate resource-types.

auto-scaling-configurations

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
AUTO_SCALING_CONFIGURATION_INSPECT	`ListAutoScalingConfigurations` `ListAutoScalingPolicies`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT +	*INSPECT* +	*none*
AUTO_SCALING_CONFIGURATION_READ	`GetAutoScalingConfiguration`	
	`GetAutoScalingPolicy`	

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	*no extra*	*USE* +
AUTO_SCALING_CONFIGURATION_CREATE		`CreateAutoScalingConfiguration`
AUTO_SCALING_CONFIGURATION_UPDATE		`UpdateAutoScalingConfiguration`
AUTO_SCALING_CONFIGURATION_DELETE		`DeleteAutoScalingConfiguration`
		`CreateAutoScalingPolicy`
		`UpdateAutoScalingPolicy`
		`DeleteAutoScalingPolicy`
		All also need `manage instance-pools`.

FOR VOLUME-FAMILY RESOURCE TYPES

The following tables list the permissions and API operations covered by each of the individual resource-types included in `volume-family`.

volumes

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
VOLUME_INSPECT	`ListVolumes` `GetVolume`	`ListVolumeBackups`, `GetVolumeBackup` (these also need `inspect volume-backups`) `UpdateVolumeBackup` (also need `read volume-backups`) `DeleteVolumeBackup` (also need `manage volume-backups`) `GetVolumeAttachment` (also need `inspect instances` and `inspect volume-attachments`). If you need to get the CHAP secret if it exists, `read volume-attachments` is required.

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*no extra*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + VOLUME_UPDATE VOLUME_WRITE	*no extra*	*READ* + `AttachVolume` and `DetachVolume` (both also need `manage volume-attachments, use instances`) `CreateVolumeBackup` (also need `manage volume-backups`)

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	*USE* +	*USE* +
VOLUME_CREATE	`CreateVolume`	If creating a volume *from a backup*, also need `read volume-backups`.
VOLUME_DELETE	`DeleteVolume`	If creating a volume *encrypted with a Key Management master encryption key*, also need `use key-delegates` (for the caller) and `read keys` (for the service principal). For more information, see Details for the Key Management Service.

volume-attachments

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
VOLUME_ATTACHMENT_INSPECT	`ListVolumeAttachments`	`GetVolumeAttachment` (also need `inspect volumes` and `inspect instances`) **Note:** The CHAP secret (if it exists) is NOT included with `inspect volume-attachments`.

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT +	*no extra*	Same as for `inspect volume-attachments`, except that `GetVolumeAttachment` also includes the CHAP secret, if it exists.
VOLUME_ATTACHMENT_READ		

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ +	*no extra*	*no extra*
VOLUME_ATTACHMENT_UPDATE		

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + VOLUME_ATTACHMENT_CREATE VOLUME_ATTACHMENT_DELETE	*no extra*	*USE* + `AttachVolume, DetachVolume` (both also need `use volumes` and `use instances`)

volume-backups

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
VOLUME_BACKUP_INSPECT	*none*	`ListVolumeBackups, GetVolumeBackup` (both also need `inspect volumes`)

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT + VOLUME_BACKUP_READ	*none*	*INSPECT* + `CreateVolume` when creating volume from an backup (also need `manage volumes`)

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + VOLUME_BACKUP_COPY VOLUME_BACKUP_UPDATE	*none*	*READ* + `UpdateVolumeBackup` (also need `inspect volumes`) `CopyVolumeBackup` (also need `create volume backups` in destination region)

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + VOLUME_BACKUP_CREATE VOLUME_BACKUP_DELETE	*none*	*USE* + `CreateVolumeBackup` (also need `use volumes`) `DeleteVolumeBackup` (also need `inspect volumes`)

boot-volume-backups

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
BOOT_VOLUME_BACKUP_INSPECT	*none*	`ListBootVolumeBackups`, `GetBootVolumeBackup` (both also need `inspect volumes`)

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT + BOOT_VOLUME_BACKUP_READ	*none*	*INSPECT* + `CreateBootVolume` when creating volume from an backup (also need `manage volumes`)

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + BOOT_VOLUME_BACKUP_UPDATE	*none*	*READ* + `UpdateBootVolumeBackup` (also need `inspect volumes`)

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	*none*	*USE* +
BOOT_VOLUME_BACKUP_CREATE		`CreateBootVolumeBackup` (also need `use volumes`)
BOOT_VOLUME_BACKUP_DELETE		`DeleteBootVolumeBackup` (also need `inspect volumes`)

backup-policies

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
BACKUP_POLICY_INSPECT	`ListVolumeBackupPolicies`	*none*
	`GetVolumeBackupPolicy`	

volume-groups

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
VOLUME_GROUP_INSPECT	`ListVolumeGroups`	*no extra*
	`GetVolumeGroup`	

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*no extra*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*no extra*

MANAGE

Permissions	APIs Fully Covered
USE +	*USE* +
VOLUME_GROUP_UPDATE	`DeleteVolumeGroup`
VOLUME_GROUP_CREATE	
VOLUME_GROUP_DELETE	

volume-group-backups

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
VOLUME_GROUP_BACKUP_INSPECT	`ListVolumeGroupBackups` `GetVolumeGroupBackup`	*no extra*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*no extra*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*no extra*

MANAGE		
Permissions	**APIs Fully Covered**	**APIs Partially Covered**
USE +	*USE* +	*USE* +
VOLUME_GROUP_BACKUP_UPDATE	`UpdateVolumeGroupBackup`	`CreateVolumeGroupBackup` also need the following: • `inspect volume group` for the source volume group • `create volume group backup` • `write volume` for the source volumes • `create volume backup` or `create boot volume backup` for the destination volumes `DeleteVolumeGroupBackup` also need `delete volume backup` or `delete boot volume backup`
VOLUME_GROUP_BACKUP_CREATE		
VOLUME_GROUP_BACKUP_DELETE		

Permissions Required for Each API Operation

The following tables list the API operations grouped by resource type. The resource types are listed in alphabetical order.

For information about permissions, see Permissions.

CORE SERVICES API OPERATIONS

API Operation	Permissions Required to Use the Operation
GetVolumeBackupPolicy	BACKUP_POLICY_INSPECT
ListVolumeBackupPolicies	BACKUP_POLICY_INSPECT
ListConsoleHistories	CONSOLE_HISTORY_READ and INSTANCE_INSPECT
GetConsoleHistory	CONSOLE_HISTORY_READ and INSTANCE_INSPECT
ShowConsoleHistoryData	CONSOLE_HISTORY_READ and INSTANCE_READ and INSTANCE_IMAGE_READ
CaptureConsoleHistory	CONSOLE_HISTORY_CREATE and INSTANCE_READ and INSTANCE_IMAGE_READ
DeleteConsoleHistory	CONSOLE_HISTORY_DELETE
ListCpes	CPE_READ
GetCpe	CPE_READ
UpdateCpe	CPE_UPDATE
CreateCpe	CPE_CREATE
DeleteCpe	CPE_DELETE
ListCrossConnects	CROSS_CONNECT_READ
GetCrossConnect	CROSS_CONNECT_READ

API Operation	Permissions Required to Use the Operation
`UpdateCrossConnect`	CROSS_CONNECT_UPDATE
`CreateCrossConnect`	CROSS_CONNECT_CREATE if not creating cross-connect in a cross-connect group. If creating the cross-connect in a cross-connect group, also need CROSS_CONNECT_CREATE and CROSS_CONNECT_ATTACH
`DeleteCrossConnect`	CROSS_CONNECT_DELETE if cross-connect is not in a cross-connect group. If the cross-connect is in a cross-connect group, also need CROSS_CONNECT_DELETE and CROSS_CONNECT_DETACH
`ListCrossConnectGroups`	CROSS_CONNECT_GROUP_READ
`GetCrossConnectGroup`	CROSS_CONNECT_GROUP_READ
`UpdateCrossConnectGroup`	CROSS_CONNECT_GROUP_UPDATE
`CreateCrossConnectGroup`	CROSS_CONNECT_GROUP_CREATE
`DeleteCrossConnectGroup`	CROSS_CONNECT_GROUP_DELETE
`ListDhcpOptions`	DHCP_READ
`GetDhcpOptions`	DHCP_READ
`UpdateDhcpOptions`	DHCP_UPDATE
`CreateDhcpOptions`	DHCP_CREATE and VCN_ATTACH

API Operation	Permissions Required to Use the Operation
`DeleteDhcpOptions`	DHCP_DELETE and VCN_DETACH
`ListDrgs`	DRG_READ
`GetDrg`	DRG_READ
`UpdateDrg`	DRG_UPDATE
`CreateDrg`	DRG_CREATE
`DeleteDrg`	DRG_DELETE
`ListDrgAttachments`	DRG_ATTACHMENT_READ
`GetDrgAttachment`	DRG_ATTACHMENT_READ
`UpdateDrgAttachment`	DRG_ATTACHMENT_UPDATE ROUTE_TABLE_ATTACH is necessary to associate a route table with the DRG attachment during the update.
`CreateDrgAttachment`	DRG_ATTACH and VCN_ATTACH ROUTE_TABLE_ATTACH is necessary to associate a route table with the DRG attachment during creation.
`DeleteDrgAttachment`	DRG_DETACH and VCN_DETACH
`CreateInstanceConsoleConnection`	INSTANCE_CONSOLE_CONNECTION_ CREATE and INSTANCE_READ
`DeleteInstanceConsoleConnection`	INSTANCE_CONSOLE_CONNECTION_ DELETE

API Operation	Permissions Required to Use the Operation
`GetInstanceConsoleConnection`	INSTANCE_CONSOLE_CONNECTION_READ and INSTANCE_READ
`ListInstanceConsoleConnections`	INSTANCE_CONSOLE_CONNECTION_ INSPECT and INSTANCE_INSPECT and INSTANCE_READ
`ListImages`	INSTANCE_IMAGE_READ
`GetImage`	INSTANCE_IMAGE_READ
`UpdateImage`	INSTANCE_IMAGE_UPDATE
`CreateImage`	INSTANCE_IMAGE_CREATE and INSTANCE_CREATE_IMAGE The first permission is related to the `instance-image`; the second is related to the `instance`.
`DeleteImage`	INSTANCE_IMAGE_DELETE
`ListInstances`	INSTANCE_READ
`GetInstance`	INSTANCE_READ
`LaunchInstance`	INSTANCE_CREATE and INSTANCE_ IMAGE_READ and VNIC_CREATE and VNIC_ATTACH and SUBNET_ATTACH
`UpdateInstance`	INSTANCE_UPDATE
`InstanceAction`	INSTANCE_POWER_ACTIONS

API Operation	Permissions Required to Use the Operation
TerminateInstance	INSTANCE_DELETE and VNIC_DELETE and SUBNET_DETACH
ListInstanceConfigurations	INSTANCE_CONFIGURATION_INSPECT
GetInstanceConfiguration	INSTANCE_CONFIGURATION_READ
LaunchInstanceConfiguration	INSTANCE_CONFIGURATION_LAUNCH
UpdateInstanceConfiguration	INSTANCE_CONFIGURATION_UPDATE
CreateInstanceConfiguration	INSTANCE_CONFIGURATION_CREATE
DeleteInstanceConfiguration	INSTANCE_CONFIGURATION_DELETE
ListInstancePools	INSTANCE_POOL_INSPECT
ListInstancePoolInstances	INSTANCE_POOL_READ
GetInstancePool	INSTANCE_POOL_READ
UpdateInstancePool	INSTANCE_POOL_UPDATE
ResetInstancePool	INSTANCE_POOL_POWER_ACTIONS
SoftresetInstancePool	INSTANCE_POOL_POWER_ACTIONS
StartInstancePool	INSTANCE_POOL_POWER_ACTIONS
StopInstancePool	INSTANCE_POOL_POWER_ACTIONS
CreateInstancePool	INSTANCE_POOL_CREATE
TerminateInstancePool	INSTANCE_POOL_DELETE
ListInternetGateways	INTERNET_GATEWAY_READ

API Operation	Permissions Required to Use the Operation
GetInternetGateway	INTERNET_GATEWAY_READ
UpdateInternetGateway	INTERNET_GATEWAY_UPDATE
CreateInternetGateway	INTERNET_GATEWAY_CREATE and VCN_ATTACH
DeleteInternetGateway	INTERNET_GATEWAY_DELETE and VCN_DETACH
ListIPSecConnections	IPSEC_CONNECTION_READ
GetIPSecConnection	IPSEC_CONNECTION_READ
UpdateIpSecConnection	IPSEC_CONNECTION_UPDATE
CreateIPSecConnection	DRG_ATTACH and CPE_ATTACH and IPSEC_CONNECTION_CREATE
DeleteIPSecConnection	DRG_DETACH and CPE_DETACH and IPSEC_CONNECTION_DELETE
GetIPSecConnectionDeviceConfig	IPSEC_CONNECTION_DEVICE_CONFIG_READ
GetIPSecConnectionDeviceStatus	IPSEC_CONNECTION_READ
ListIPSecConnectionTunnels	IPSEC_CONNECTION_READ
GetIPSecConnectionTunnel	IPSEC_CONNECTION_READ
UpdateIPSecConnectionTunnel	IPSEC_CONNECTION_UPDATE

API Operation	Permissions Required to Use the Operation
`GetIPSecConnectionTunnelSharedSecret`	IPSEC_CONNECTION_DEVICE_CONFIG_READ
`UpdateIPSecConnectionTunnelSharedSecret`	IPSEC_CONNECTION_DEVICE_CONFIG_UPDATE
`ListLocalPeeringGateways`	LOCAL_PEERING_GATEWAY_READ
`GetLocalPeeringGateway`	LOCAL_PEERING_GATEWAY_READ
`UpdateLocalPeeringGateway`	LOCAL_PEERING_GATEWAY_UPDATE ROUTE_TABLE_ATTACH is necessary to associate a route table with the LPG during the update.
`CreateLocalPeeringGateway`	LOCAL_PEERING_GATEWAY_CREATE and VCN_ATTACH ROUTE_TABLE_ATTACH is necessary to associate a route table with the LPG during creation.
`DeleteLocalPeeringGateway`	LOCAL_PEERING_GATEWAY_DELETE and VCN_DETACH
`ConnectLocalPeeringGateway`	LOCAL_PEERING_GATEWAY_CONNECT_FROM and LOCAL_PEERING_GATEWAY_CONNECT_TO
`ListNatGateways`	NAT_GATEWAY_READ
`GetNatGateway`	NAT_GATEWAY_READ

API Operation	Permissions Required to Use the Operation
UpdateNatGateway	NAT_GATEWAY_UPDATE
CreateNatGateway	NAT_GATEWAY_CREATE and VCN_READ and VCN_ATTACH
DeleteNatGateway	NAT_GATEWAY_DELETE and VCN_READ and VCN_DETACH
ListPrivateIps	PRIVATE_IP_READ
GetPrivateIp	PRIVATE_IP_READ
UpdatePrivateIp	PRIVATE_IP_UPDATE
CreatePrivateIp	PRIVATE_IP_CREATE and PRIVATE_IP_ ASSIGN and VNIC_ASSIGN and SUBNET_ ATTACH
DeletePrivateIp	PRIVATE_IP_DELETE and PRIVATE_IP_ UNASSIGN and VNIC_UNASSIGN and SUBNET_DETACH
ListRemotePeeringConnections	REMOTE_PEERING_CONNECTION_READ
GetRemotePeeringConnection	REMOTE_PEERING_CONNECTION_READ
UpdateRemotePeeringConnection	REMOTE_PEERING_CONNECTION_UPDATE
CreateRemotePeeringConnection	REMOTE_PEERING_CONNECTION_CREATE and DRG_ATTACH
DeleteRemotePeeringConnection	REMOTE_PEERING_CONNECTION_DELETE and DRG_DETACH

API Operation	Permissions Required to Use the Operation
ConnectRemotePeeringConnections	REMOTE_PEERING_CONNECTION_CONNECT_FROM and REMOTE_PEERING_CONNECTION_CONNECT_TO
ListPublicIps	For ephemeral public IPs: PRIVATE_IP_READ For reserved public IPs: PUBLIC_IP_READ
GetPublicIp	For ephemeral public IPs: PRIVATE_IP_READ For reserved public IPs: PUBLIC_IP_READ
GetPublicIpByPrivateIpId	For ephemeral public IPs: PRIVATE_IP_READ For reserved public IPs: PUBLIC_IP_READ
GetPublicIpByIpAddress	For ephemeral public IPs: PRIVATE_IP_READ For reserved public IPs: PUBLIC_IP_READ
UpdatePublicIP	For ephemeral public IPs: PRIVATE_IP_UPDATE For reserved public IPs: PUBLIC_IP_UPDATE and PRIVATE_IP_ASSIGN_PUBLIC_IP and PUBLIC_IP_ASSIGN_PRIVATE_IP and PRIVATE_IP_UNASSIGN_PUBLIC_IP and PUBLIC_IP_UNASSIGN_PRIVATE_IP

API Operation	Permissions Required to Use the Operation
`CreatePublicIp`	For ephemeral public IPs: PRIVATE_IP_ASSIGN_PUBLIC_IP For reserved public IPs: PUBLIC_IP_CREATE and PUBLIC_IP_ASSIGN_PRIVATE_IP and PRIVATE_IP_ASSIGN_PUBLIC_IP
`DeletePublicIp`	For ephemeral public IPs: PRIVATE_IP_UNASSIGN_PUBLIC_IP For reserved public IPs: PUBLIC_IP_DELETE and PUBLIC_IP_UNASSIGN_PRIVATE_IP and PRIVATE_IP_UNASSIGN_PUBLIC_IP
`ListRouteTables`	ROUTE_TABLE_READ
`GetRouteTable`	ROUTE_TABLE_READ

API Operation	Permissions Required to Use the Operation
`UpdateRouteTable`	ROUTE_TABLE_UPDATE and INTERNET_GATEWAY_ATTACH (if creating a route rule that uses an internet gateway as a target) and INTERNET_GATEWAY_DETACH (if deleting a route rule that uses an internet gateway as a target) and DRG_ATTACH (if creating a route rule that uses a DRG as a target) and DRG_DETACH (if deleting a route rule that uses a DRG as a target) and PRIVATE_IP_ROUTE_TABLE_ATTACH (if creating a route rule that uses a private IP as a target) and PRIVATE_IP_ROUTE_TABLE_DETACH (if deleting a route rule that uses a private IP as a target) and LOCAL_PEERING_GATEWAY_ATTACH (if creating a route rule that uses an LPG as a target) and LOCAL_PEERING_GATEWAY_DETACH (if deleting a route rule that uses an LPG as a target) and NAT_GATEWAY_ATTACH (if creating a route rule that uses a NAT gateway as a target) and

API Operation	Permissions Required to Use the Operation
	NAT_GATEWAY_DETACH (if deleting a route rule that uses a NAT gateway as a target) and SERVICE_GATEWAY_ATTACH (if creating a route rule that uses a service gateway as a target) and SERVICE_GATEWAY_DETACH (if deleting a route rule that uses a service gateway as a target)

API Operation	Permissions Required to Use the Operation
`CreateRouteTable`	ROUTE_TABLE_CREATE and VCN_ATTACH and INTERNET_GATEWAY_ATTACH (if creating a route rule that uses an internet gateway as a target) and DRG_ATTACH (if creating a route rule that uses a DRG as a target) and PRIVATE_IP_ROUTE_TABLE_ATTACH (if creating a route rule that uses a private IP as a target) and LOCAL_PEERING_GATEWAY_ATTACH (if creating a route rule that uses an LPG as a target) and NAT_GATEWAY_ATTACH (if creating a route rule that uses a NAT gateway as a target) and SERVICE_GATEWAY_ATTACH (if creating a route rule that uses a service gateway as a target)

API Operation	Permissions Required to Use the Operation
`DeleteRouteTable`	ROUTE_TABLE_DELETE and VCN_DETACH and INTERNET_GATEWAY_DETACH (if deleting a route rule that uses an internet gateway as a target) and DRG_DETACH (if deleting a route rule that uses a DRG as a target) and PRIVATE_IP_ROUTE_TABLE_DETACH (if deleting a route rule that uses a private IP as a target) and LOCAL_PEERING_GATEWAY_DETACH (if deleting a route rule that uses an LPG as a target) and NAT_GATEWAY_DETACH (if deleting a route rule that uses a NAT gateway as a target) and SERVICE_GATEWAY_DETACH (if deleting a route rule that uses a service gateway as a target)
`ListSecurityLists`	SECURITY_LIST_READ
`GetSecurityList`	SECURITY_LIST_READ
`UpdateSecurityList`	SECURITY_LIST_UPDATE
`CreateSecurityList`	SECURITY_LIST_CREATE and VCN_ ATTACH

API Operation	Permissions Required to Use the Operation
DeleteSecurityList	SECURITY_LIST_DELETE and VCN_DETACH
ListServiceGateways	SERVICE_GATEWAY_READ
GetServiceGateway	SERVICE_GATEWAY_READ
UpdateServiceGateway	SERVICE_GATEWAY_UPDATE
CreateServiceGateway	SERVICE_GATEWAY_CREATE and VCN_ATTACH
DeleteServiceGateway	SERVICE_GATEWAY_DELETE and VCN_DETACH
AttachServiceId	SERVICE_GATEWAY_ADD_SERVICE
DetachServiceId	SERVICE_GATEWAY_DELETE_SERVICE
ListShapes	MACHINE_SHAPE_READ
ListSubnets	SUBNET_READ
GetSubnet	SUBNET_READ

API Operation	Permissions Required to Use the Operation
`UpdateSubnet`	SUBNET_UPDATE If changing which route table is associated with the subnet, also need ROUTE_TABLE_ATTACH and ROUTE_TABLE_DETACH If changing which security lists are associated with the subnet, also need SECURITY_LIST_ATTACH and SECURITY_LIST_DETACH If changing which set of DHCP options are associated with the subnet, also need DHCP_ATTACH and DHCP_DETACH
`CreateSubnet`	SUBNET_CREATE and VCN_ATTACH and ROUTE_TABLE_ATTACH and SECURITY_LIST_ATTACH and DHCP_ATTACH
`DeleteSubnet`	SUBNET_DELETE and VCN_DETACH and ROUTE_TABLE_DETACH and SECURITY_LIST_DETACH and DHCP_DETACH
`ListVcns`	VCN_READ
`GetVcn`	VCN_READ
`UpdateVcn`	VCN_UPDATE
`CreateVcn`	VCN_CREATE
`DeleteVcn`	VCN_DELETE
`ListVirtualCircuits`	VIRTUAL_CIRCUIT_READ

API Operation	Permissions Required to Use the Operation
`GetVirtualCircuit`	VIRTUAL_CIRCUIT_READ
`UpdateVirtualCircuit`	VIRTUAL_CIRCUIT_UPDATE and DRG_ATTACH and DRG_DETACH If updating which cross-connect or cross-connect group the virtual circuit is using, also need CROSS_CONNECT_DETACH and CROSS_CONNECT_ATTACH
`CreateVirtualCircuit`	VIRTUAL_CIRCUIT_CREATE and DRG_ATTACH If creating the virtual circuit with a mapping to a specific cross-connect or cross-connect group, also need CROSS_CONNECT_ATTACH
`DeleteVirtualCircuit`	VIRTUAL_CIRCUIT_DELETE and DRG_DETACH If deleting a virtual circuit that's currently using a cross-connect or cross-connect group, also need CROSS_CONNECT_DETACH
`GetVnic`	VNIC_READ
`AttachVnic`	INSTANCE_ATTACH_SECONDARY_VNIC and VNIC_ATTACH and VNIC_CREATE and SUBNET_ATTACH

API Operation	Permissions Required to Use the Operation
`DetachVnic`	INSTANCE_DETACH_SECONDARY_VNIC and VNIC_DETACH and VNIC_DELETE and SUBNET_DETACH
`UpdateVnic`	VNIC_UPDATE
`ListVnicAttachments`	VNIC_ATTACHMENT_READ and INSTANCE_INSPECT
`GetVnicAttachment`	VNIC_ATTACHMENT_READ
`ListVolumes`	VOLUME_INSPECT
`GetVolume`	VOLUME_INSPECT
`UpdateVolume`	VOLUME_UPDATE
`CreateVolume`	VOLUME_CREATE (and VOLUME_BACKUP_READ if creating volume from a backup)
`DeleteVolume`	VOLUME_DELETE
`ListVolumeAttachments`	VOLUME_ATTACHMENT_INSPECT and VOLUME_INSPECT and INSTANCE_INSPECT
`GetVolumeAttachment`	VOLUME_ATTACHMENT_INSPECT and VOLUME_INSPECT and INSTANCE_INSPECT **Note:** To also get the CHAP secret for the volume, then VOLUME_ATTACHMENT_READ is required instead of VOLUME_ATTACHMENT_INSPECT

API Operation	Permissions Required to Use the Operation
`AttachVolume`	VOLUME_ATTACHMENT_CREATE and VOLUME_WRITE and INSTANCE_ATTACH_VOLUME
`DetachVolume`	VOLUME_ATTACHMENT_DELETE and VOLUME_WRITE and INSTANCE_DETACH_VOLUME
`ListVolumeBackups`	VOLUME_BACKUP_INSPECT and VOLUME_INSPECT
`GetVolumeBackup`	VOLUME_BACKUP_INSPECT and VOLUME_INSPECT
`UpdateVolumeBackup`	VOLUME_BACKUP_UPDATE and VOLUME_INSPECT
`CreateVolumeBackup`	VOLUME_BACKUP_CREATE and VOLUME_WRITE
`DeleteVolumeBackup`	VOLUME_BACKUP_DELETE and VOLUME_INSPECT
`CreateBootVolume`	VOLUME_CREATE, BOOT_VOLUME_BACKUP_READ, VOLUME_WRITE
`GetBootVolume`	VOLUME_INSPECT
`ListBootVolumes`	VOLUME_INSPECT
`UpdateBootVolume`	VOLUME_UPDATE
`DeleteBootVolume`	VOLUME_DELETE

API Operation	Permissions Required to Use the Operation
`CreateBootVolumeBackup`	BOOT_VOLUME_BACKUP_CREATE, VOLUME_WRITE
`ListBootVolumeBackups`	BOOT_VOLUME_BACKUP_INSPECT, VOLUME_INSPECT
`GetBootVolumeBackup`	BOOT_VOLUME_BACKUP_INSPECT, VOLUME_INSPECT
`UpdateBootVolumeBackup`	BOOT_VOLUME_BACKUP_UPDATE, VOLUME_INSPECT
`DeleteBootVolumeBackup`	BOOT_VOLUME_BACKUP_DELETE, VOLUME_INSPECT
`CreateVolumeGroup`	VOLUME_GROUP_CREATE, VOLUME_INSPECT if creating the volume group from a list of volumes. VOLUME_GROUP_CREATE, VOLUME_GROUP_INSPECT, VOLUME_CREATE, VOLUME_WRITE if cloning a volume group. VOLUME_GROUP_CREATE, VOLUME_GROUP_BACKUP_INSPECT, VOLUME_BACKUP_READ/BOOT_VOLUME_BACKUP_READ, VOLUME_CREATE, VOLUME_WRITE if restoring from a volume group backup.
`DeleteVolumeGroup`	VOLUME_GROUP_DELETE
`GetVolumeGroup`	VOLUME_GROUP_INSPECT

API Operation	Permissions Required to Use the Operation
`ListVolumeGroups`	VOLUME_GROUP_INSPECT
`UpdateVolumeGroup`	VOLUME_GROUP_UPDATE, VOLUME_INSPECT
`CreateVolumeGroupBackup`	VOLUME_GROUP_BACKUP_CREATE, VOLUME_GROUP_INSPECT, VOLUME_WRITE, VOLUME_BACKUP_CREATE/BOOT_VOLUME_BACKUP_CREATE
`DeleteVolumeGroupBackup`	VOLUME_GROUP_BACKUP_DELETE, VOLUME_BACKUP_DELETE/BOOT_VOLUME_BACKUP_DELETE
`GetVolumeGroupBackup`	VOLUME_GROUP_BACKUP_INSPECT
`ListVolumeGroupBackups`	VOLUME_GROUP_BACKUP_INSPECT
`UpdateVolumeGroupBackup`	VOLUME_GROUP_BACKUP_UPDATE

AUTOSCALING API OPERATIONS

API Operation	Permissions Required to Use the Operation
`ListAutoScalingConfigurations`	AUTO_SCALING_CONFIGURATION_INSPECT
`GetAutoScalingConfiguration`	AUTO_SCALING_CONFIGURATION_READ
`UpdateAutoScalingConfiguration`	AUTO_SCALING_CONFIGURATION_UPDATE and INSTANCE_POOL_UPDATE
`CreateAutoScalingConfiguration`	AUTO_SCALING_CONFIGURATION_CREATE and INSTANCE_POOL_UPDATE

API Operation	Permissions Required to Use the Operation
`DeleteAutoScalingConfiguration`	AUTO_SCALING_CONFIGURATION_DELETE and INSTANCE_POOL_UPDATE
`ListAutoScalingPolicies`	AUTO_SCALING_CONFIGURATION_READ
`GetAutoScalingPolicy`	AUTO_SCALING_CONFIGURATION_READ
`UpdateAutoScalingPolicy`	AUTO_SCALING_CONFIGURATION_UPDATE and INSTANCE_POOL_UPDATE
`CreateAutoScalingPolicy`	AUTO_SCALING_CONFIGURATION_CREATE and INSTANCE_POOL_UPDATE
`DeleteAutoScalingPolicy`	AUTO_SCALING_CONFIGURATION_DELETE and INSTANCE_POOL_UPDATE

Details for Container Engine for Kubernetes

This topic covers details for writing policies to control access to Container Engine for Kubernetes.

Resource-Types

Aggregate Resource-Type

- `cluster-family`

Individual Resource-Types

- `clusters`
- `cluster-node-pools`
- `cluster-work-requests`

COMMENTS

A policy that uses `<verb> cluster-family` is equivalent to writing one with a separate `<verb> <individual resource-type>` statement for each of the individual resource-types.

See the table in Details for Verb + Resource-Type Combinations for a detailed breakout of the API operations covered by each verb, for each individual resource-type included in `cluster-family`.

Supported Variables

Container Engine for Kubernetes supports all the general variables (see General Variables for All Requests), plus the ones listed here.

The `clusters` resource type can use the following variables:

Variable	Variable Type	Comments
`target.cluster.id`	Entity (OCID)	

The `cluster-node-pools` resource type can use the following variables:

Variable	Variable Type	Comments
`target.nodepool.id`	Entity (OCID)	

Details for Verb + Resource-Type Combinations

The following tables show the permissions and API operations covered by each verb. The level of access is cumulative as you go from `inspect` > `read` > `use` > `manage`. A plus sign (+) in a table cell indicates incremental access compared to the cell directly above it, whereas "no extra" indicates no incremental access..

For example, the `read` verb for the `clusters` resource-type includes the same permissions and API operations as the `inspect` verb, plus the CLUSTER_READ permission and a number of API operations (e.g., `GetCluster`, etc.). The `use` verb covers still another permission and API

operation compared to `read`. Lastly, `manage` covers more permissions and operations compared to `use`.

clusters

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
CLUSTER_INSPECT	`ListClusters`	*none*
	`ListWorkRequests`	

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT +	*INSPECT* +	*none*
CLUSTER_READ	`GetCluster`	
	`GetWorkRequest`	
	`ListWorkRequestErrors`	
	`ListWorkRequestLogs`	

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ +	*READ* +	*none*
CLUSTER_USE	`GetClusterKubeconfig`	

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	*USE* +	*none*
CLUSTER_CREATE	`CreateCluster`	
CLUSTER_DELETE	`DeleteCluster`	
CLUSTER_UPDATE	`UpdateCluster`	
CLUSTER_MANAGE	`AdministerK8s`	

cluster-node-pools

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
CLUSTER_NODE_POOL_INSPECT	`ListNodePools` `ListWorkRequests`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT + CLUSTER_NODE_POOL_READ	*INSPECT* + `GetNodePool` `GetWorkRequest` `ListWorkRequestErrors` `ListWorkRequestLogs`	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + CLUSTER_NODE_POOL_CREATE CLUSTER_NODE_POOL_DELETE CLUSTER_NODE_POOL_UPDATE	*USE* + `CreateNodePool` `DeleteNodePool` `UpdateNodePool`	*none*

cluster-work-requests

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
CLUSTER_WORK_REQUEST_INSPECT	`ListWorkRequests`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT +	*INSPECT* +	*none*
CLUSTER_WORK_REQUEST_READ	`GetWorkRequest`	
	`ListWorkRequestErrors`	
	`ListWorkRequestLogs`	

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	*USE* +	*none*
CLUSTER_WORK_REQUEST_DELETE	`DeleteWorkRequest`	

Permissions Required for Each API Operation

The following table lists the API operations in a logical order, grouped by resource type. For information about permissions, see Permissions.

API Operation	Permissions Required to Use the Operation
`ListClusters`	CLUSTER_INSPECT
`CreateCluster`	CLUSTER_CREATE
`GetClusterKubeconfig`	CLUSTER_USE
`GetCluster`	CLUSTER_READ
`UpdateCluster`	CLUSTER_UPDATE
`DeleteCluster`	CLUSTER_DELETE

API Operation	Permissions Required to Use the Operation
AdministerK8s	CLUSTER_MANAGE
ListNodePools	CLUSTER_NODE_POOL_INSPECT
CreateNodePool	CLUSTER_NODE_POOL_CREATE
GetNodePool	CLUSTER_NODE_POOL_READ
UpdateNodePool	CLUSTER_NODE_POOL_UPDATE
DeleteNodePool	CLUSTER_NODE_POOL_DELETE
ListWorkRequests	CLUSTER_WORK_REQUEST_INSPECT, CLUSTER_NODE_POOL_INSPECT, CLUSTER_INSPECT
GetWorkRequest	CLUSTER_WORK_REQUEST_READ, CLUSTER_NODE_POOL_READ, CLUSTER_READ
ListWorkRequestErrors	CLUSTER_WORK_REQUEST_READ, CLUSTER_NODE_POOL_READ, CLUSTER_READ
ListWorkRequestLogs	CLUSTER_WORK_REQUEST_READ, CLUSTER_NODE_POOL_READ, CLUSTER_READ
DeleteWorkRequest	CLUSTER_WORK_REQUEST_DELETE

Details for the Database Service

This topic covers details for writing policies to control access to the Database service.

Resource-Types

`database-family`, which covers these individual resource-types:

`db-systems`

- `db-nodes`
- `db-homes`
- `databases`
- `backups`

`autonomous-database-family`, which covers these individual resource-types:

- `autonomous-database`
- `autonomous-backup`



Tip

See sample policies for Autonomous Database in Let database admins manage Autonomous Databases. You have the option of limiting a policy to either the Autonomous Data Warehouse or Autonomous Transaction Processing workload type if needed.

Supported Variables

Only the general variables are supported (see General Variables for All Requests).

Details for Verb + Resource-Type Combinations

The following tables show the permissions and API operations covered by each verb. The level of access is cumulative as you go from `inspect` > `read` > `use` > `manage`. A plus sign (+) in a table cell indicates incremental access compared to the cell directly above it, whereas "no extra" indicates no incremental access.

For example, the `read` and `use` verbs for the `db-systems` resource-type cover no extra permissions or API operations compared to the `inspect` verb. However, the `manage` verb includes two more permissions and partially covers two more API operations.

FOR DATABASE-FAMILY RESOURCE TYPES

db-systems

INSPECT

Permission	APIs Fully Covered	APIs Partially Covered
DB_SYSTEM_INSPECT	`ListDbSystems` `GetDbSystem` `ListDbSystemPatches` `ListDbSystemPatchHistoryEntries` `GetDbSystemPatch` `GetDbSystemPatchHistoryEntry`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + DB_SYSTEM_UPDATE DB_SYSTEM_CREATE DB_SYSTEM_DELETE	`UpdateDBSystem`	`LaunchDBSystem`, `TerminateDbSystem` (both also need `manage db-homes`, `manage databases`, `use vnics`, and `use subnets`)

db-nodes

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
DB_NODE_INSPECT	`GetDbNode`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	*USE* +	*none*
DB_NODE_POWER_ACTIONS	`DbNodeAction`	

db-homes

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
DB_HOME_INSPECT	`ListDBHome`	*none*
	`GetDBHome`	
	`ListDbHomePatches`	
	`ListDbHomePatchHistoryEntries`	
	`GetDbHomePatch`	
	`GetDbHomePatchHistoryEntry`	

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + DB_HOME_CREATE DB_HOME_UPDATE DB_HOME_DELETE	`UpdateDBHome`	`LaunchDBSystem`, `TerminateDbSystem` (both also need `manage db-systems`, `manage databases`, `use vnics`, and `use subnets`)

databases

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
DATABASE_INSPECT	`ListDatabases` `GetDatabase` `ListDataGuardAssociations` `GetDataGuardAssociation`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + DATABASE_CREATE DATABASE_DELETE	*no extra*	`LaunchDBSystem`, `TerminateDbSystem` (both also need `manage db-systems`, `manage db-homes`, `use vnics`, and `use subnets`)

FOR AUTONOMOUS-TRANSACTION-PROCESSING-FAMILY RESOURCE TYPES

autonomous-databases

INSPECT

Permissions	APIs Fully Covered	
AUTONOMOUS_DATABASE_INSPECT	`GetAutonomousDatabase`, `ListAutonomousDatabases`	*no extra*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT + AUTONOMOUS_DATABASE_CONTENT_READ	*no extra*	`CreateAutonomousDatabaseBackup` (also needs `manage autonomous-backups`)

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + AUTONOMOUS_DATABASE_CONTENT_WRITE AUTONOMOUS_DATABASE_UPDATE	`UpdateAutonomousDatabase`	`RestoreAutonomousDatabase` (also needs `read autonomous-backups`)

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + AUTONOMOUS_DATABASE_CREATE AUTONOMOUS_DATABASE_DELETE	*no extra*	`CreateAutonomousDatabase`

autonomous-backups

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
AUTONOMOUS_DB_BACKUP_INSPECT	`ListAutonomousDatabaseBackups`, `GetAutonomousDatabaseBackup`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT + AUTONOMOUS_DB_BACKUP_CONTENT_READ	*no extra*	`RestoreAutonomousDatabase` (also needs `use autonomous-databases`)

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + AUTONOMOUS_DB_BACKUP_CREATE AUTONOMOUS_DB_BACKUP_DELETE	`DeleteAutonomousDatabaseBackup`	`CreateAutonomousDatabaseBackup` (also needs `read autonomous-databases`)

FOR AUTONOMOUS-DATA-WAREHOUSE-FAMILY RESOURCE TYPES

autonomous-data-warehouses

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
AUTONOMOUS_DW_INSPECT	`GetAutonomousDataWarehouse` `ListAutonomousDataWarehouses`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT + AUTONOMOUS_DW_CONTENT_READ	*no extra*	`CreateAutonomousDataWarehouseBackup` (also requires `manage autonomous-data-warehouse-backups`)

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + AUTONOMOUS_DW_CONTENT_WRITE AUTONOMOUS_DW_UPDATE	`UpdateAutonomousDataWarehouse` `StartAutonomousDataWarehouse` `StopAutonomousDataWarehouse`	`RestoreAutonomousDataWarehouse` (also requires `read autonomous-data-warehouse-backups`)

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + AUTONOMOUS_DW_CREATE AUTONOMOUS_DW_DELETE	`CreateAutonomousDataWarehouse` `DeleteAutonomousDataWarehouse`	*none*

autonomous-data-warehouse-backups

INSPECT

Permission	APIs Fully Covered	APIs Partially Covered
AUTONOMOUS_DW_BACKUP_INSPECT	`ListAutonomousDataWarehouseBackups` `GetAutonomousDataWarehouseBackup`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT + AUTONOMOUS_DW_BACKUP_CONTENT_READ	*no extra*	`RestoreAutonomousDataWarehouse` (also requires `use autonomous-data-warehouses`)

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + AUTONOMOUS_DW_BACKUP_CREATE	no extra	`CreateAutonomousDataWarehouseBackup` (also requires `read autonomous-data-warehouses`)

Permissions Required for Each API Operation

The following tables list the API operations for database products in a logical order, grouped by resource type.

For information about permissions, see Permissions.

DATABASE API OPERATIONS

API Operation	Permissions Required to Use the Operation
`ListDbSystems`	DB_SYSTEM_INSPECT
`GetDbSystem`	DB_SYSTEM_INSPECT
`LaunchDbSystem`	DB_SYSTEM_CREATE and DB_HOME_CREATE and DATABASE_CREATE and VNIC_CREATE and VNIC_ATTACH and SUBNET_ATTACH To enable automatic backups for the initial database, also need DB_BACKUP_CREATE and DATABASE_CONTENT_READ
`UpdateDbSystem`	DB_SYSTEM_INSPECT and DB_SYSTEM_UPDATE
`ListDbSystemPatches`	DB_SYSTEM_INSPECT
`ListDbSystemPatchHistoryEntries`	DB_SYSTEM_INSPECT

API Operation	Permissions Required to Use the Operation
GetDbSystemPatch	DB_SYSTEM_INSPECT
GetDbSystemPatchHistoryEntry	DB_SYSTEM_INSPECT
TerminateDbSystem	DB_SYSTEM_DELETE and DB_HOME_DELETE and DATABASE_DELETE and VNIC_DETACH and VNIC_DELETE and SUBNET_DETACH If automatic backups are enabled for any database in the DB System, also need DELETE_BACKUP
GetDbNode	DB_NODE_INSPECT
DbNodeAction	DB_NODE_POWER_ACTIONS
ListDbHomes	DB_HOME_INSPECT
GetDbHome	DB_HOME_INSPECT
ListDbHomePatches	DB_HOME_INSPECT
ListDbHomePatchHistoryEntries	DB_HOME_INSPECT
GetDbHomePatch	DB_HOME_INSPECT
GetDbHomePatchHistoryEntry	DB_HOME_INSPECT
CreateDbHome	DB_SYSTEM_INSPECT and DB_SYSTEM_UPDATE and DB_HOME_CREATE and DATABASE_CREATE To enable automatic backups for the database, also need DB_BACKUP_CREATE and DATABASE_CONTENT_READ
UpdateDbHome	DB_HOME_UPDATE

API Operation	Permissions Required to Use the Operation
DeleteDbHome	DB_SYSTEM_UPDATE and DB_HOME_DELETE and DATABASE_DELETE If automatic backups are enabled, also need DELETE_BACKUP If performing a final backup on termination, also need DB_BACKUP_CREATE and DATABASE_CONTENT_READ
ListDatabases	DATABASE_INSPECT
GetDatabase	DATABASE_INSPECT
UpdateDatabase	DATABASE_UPDATE To enable automatic backups, also need DB_BACKUP_CREATE and DATABASE_CONTENT_READ
ListDbSystemShapes	(no permissions required; available to anyone)
ListDbVersions	(no permissions required; available to anyone)
GetDataGuardAssociation	DATABASE_INSPECT
ListDataGuardAssociations	DATABASE_INSPECT
CreateDataGuardAssociation	DB_SYSTEM_UPDATE and DB_HOME_CREATE and DB_HOME_UPDATE and DATABASE_CREATE and DATABASE_UPDATE
SwitchoverDataGuardAssociation	DATABASE_UPDATE
FailoverDataGuardAssociation	DATABASE_UPDATE
ReinstateDataGuardAssociation	DATABASE_UPDATE

API Operation	Permissions Required to Use the Operation
GetBackup	DB_BACKUP_INSPECT
ListBackups	DB_BACKUP_INSPECT
CreateBackup	DB_BACKUP_CREATE and DATABASE_CONTENT_READ
DeleteBackup	DB_BACKUP_DELETE and DB_BACKUP_INSPECT
RestoreDatabase	DB_BACKUP_INSPECT and DB_BACKUP_CONTENT_READ and DATABASE_CONTENT_WRITE

AUTONOMOUS TRANSACTION PROCESSING API OPERATIONS

API Operation	Permissions Required to Use the Operation
GetAutonomousDatabase	AUTONOMOUS_DATABASE_INSPECT
ListAutonomousDatabases	AUTONOMOUS_DATABASE_INSPECT
CreateAutonomousDatabase	AUTONOMOUS_DATABASE_CREATE
UpdateAutonomousDatabase	AUTONOMOUS_DATABASE_UPDATE
DeleteAutonomousDatabase	AUTONOMOUS_DATABASE_DELETE
StartAutonomousDatabase	AUTONOMOUS_DATABASE_UPDATE
StopAutonomousDatabase	AUTONOMOUS_DATABASE_UPDATE
RestoreAutonomousDatabase	AUTONOMOUS_DB_BACKUP_CONTENT_READ and AUTONOMOUS_DATABASE_CONTENT_WRITE
CreateAutonomousDatabaseBackup	AUTONOMOUS_DB_BACKUP_CREATE and AUTONOMOUS_DATABASE_CONTENT_READ

API Operation	Permissions Required to Use the Operation
ListAutonomousDatabaseBackups	AUTONOMOUS_DB_BACKUP_INSPECT
GetAutonomousDatabaseBackup	AUTONOMOUS_DB_BACKUP_INSPECT

AUTONOMOUS DATA WAREHOUSE API OPERATIONS

API Operation	Permissions Required to Use the Operation
GetAutonomousDataWarehouse	AUTONOMOUS_DW_INSPECT
ListAutonomousDataWarehouses	AUTONOMOUS_DW_INSPECT
CreateAutonomousDataWarehouse	AUTONOMOUS_DW_CREATE
UpdateAutonomousDataWarehouse	AUTONOMOUS_DW_UPDATE
DeleteAutonomousDataWarehouse	AUTONOMOUS_DW_DELETE
StartAutonomousDataWarehouse	AUTONOMOUS_DW_UPDATE
StopAutonomousDataWarehouse	AUTONOMOUS_DW_UPDATE
RestoreAutonomousDataWarehouse	AUTONOMOUS_DW_BACKUP_CONTENT_READ and AUTONOMOUS_DW_CONTENT_WRITE
ListAutonomousDataWarehouseBackups	AUTONOMOUS_DW_BACKUP_INSPECT
GetAutonomousDataWarehouseBackup	AUTONOMOUS_DW_BACKUP_INSPECT
CreateAutonomousDataWarehouseBackup	AUTONOMOUS_DW_BACKUP_CREATE and AUTONOMOUS_DW_CONTENT_READ

Details for the DNS Service

This topic covers details for writing policies to control access to the DNS service.

Aggregate Resource-Type

`dns`

Individual Resource-Types

`dns-zones`

`dns-records`

`dns-traffic`

`dns-steering-policies`

`dns-steering-policy-attachments`

COMMENTS

A policy that uses `<verb> dns` is equivalent to writing one with a separate `<verb> <individual resource-type>` statement for each of the individual resource-types.

See the table in Details for Verb + Resource-Type Combinations for a detailed breakout of the API operations covered by each verb, for each individual resource-type included in `dns`.

Supported Variables

The DNS Service supports all the general variables (see General Variables for All Requests), plus the ones listed here.

The `dns-zones` resource type can use the following variables:

Variable	Variable Type	Comments
`target.dns-zone.id`	Entity (OCID)	Use this variable to control access to specific DNS zones by OCID.
`target.dns-zone.name`	String	Use this variable to control access to specific DNS zones by name.

The `dns-records` resource type can use the following variables:

Variable	Variable Type	Comments
`target.dns-zone.id`	Entity (OCID)	Use this variable to control access to specific DNS zones by OCID.
`target.dns-zone.name`	String	Use this variable to control access to specific DNS zones by name.
`target.dns-zone.scope`	String	Valid values are "public" and "private".
`target.dns-record.type`	List (String)	Use this variable to control access to specific DNS records by type. Valid values in the last can be any supported DNS resource type. For example, "A", "AAAA", "TXT", and so on. See .
`target.dns-domain.name`	List (String)	Use this variable to control access to specific domain names. Applicable to the following API operations: • `GetDomainRecords` • `PatchDomainRecords` • `UpdateDomainRecords` • `DeleteRRSet` • `GetRRSet` • `PatchRRSet` • `UpdateRRSet`

The `dns-steering-policies` resource type can use the following variables:

Variable	Variable Type	Comments
`target.dns-steering-policy.id`	Entity (OCID)	Use this variable to control access to specific steering policies by OCID.
`target.dns-steering-policy.display-name`	String	Use this variable to control access to specific steering policies by name.

Details for Verb + Resource-Type Combinations

The following tables show the permissions and API operations covered by each verb. The level of access is cumulative as you go from `inspect` > `read` > `use` > `manage`. A plus sign (+) in a table cell indicates incremental access compared to the cell directly above it, whereas "no extra" indicates no incremental access.

For example, the `use` and `manage` verbs for the `dns-traffic` resource-type cover no extra permissions or API operations compared to the `read` verb.

dns-zones

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
DNS_ZONE_INSPECT	`ListZones`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT + DNS_ZONE_READ	`GetZone`	`GetZoneRecords`

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ +	`UpdateZone`	`UpdateZoneRecords`
DNS_ZONE_UPDATE		`PatchZoneRecords`
		`CreateSteeringPolicyAttachment`
		`DeleteSteeringPolicyAttachment`

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
UPDATE +	`CreateZone`	*none*
DNS_ZONE_CREATE	`DeleteZone`	
DNS_ZONE_DELETE		

dns-records

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
DNS_RECORD_INSPECT	*none*	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT +	`GetDomainRecords`	`GetZoneRecords`
DNS_RECORD_READ	`GetRRSet`	

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ +	`PatchDomainRecords`	`UpdateZoneRecords`
DNS_RECORD_UPDATE	`UpdateDomainRecords`	`PatchZoneRecords`
	`DeleteRRSet`	`UpdateSteeringPolicyAttachment`
	`PatchRRSet`	
	`UpdateRRSet`	

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
UPDATE +	*no extra*	*none*
DNS_RECORD_CREATE		
DNS_RECORD_DELETE		

dns-traffic

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
none	*none*	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT +	`GetDNSTrafficCounts`	*none*
DNS_TRAFFIC_READ		

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

dns-steering-policies

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
DNS_STEERING_POLICY_INSPECT	`ListSteeringPolicies`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT + DNS_STEERING_POLICY_READ	GetSteeringPolicy	CreateSteeringPolicyAttachment UpdateSteeringPolicyAttachment DeleteSteeringPolicyAttachment

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + DNS_POLICY_STEERING_UPDATE	UpdateSteeringPolicy	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
UPDATE + DNS_STEERING_POLICY_CREATE DNS_STEERING_POLICY_DELETE	CreateSteeringPolicy DeleteSteeringPolicy	*none*

dns-steering-policy-attachments

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
DNS_STEERING_ATTACHMENT_INSPECT	ListSteeringPolicyAttachments	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT + DNS_STEERING_ATTACHMENT_READ	GetSteeringPolicyAttachment	*none*

Permissions Required for Each API Operation

The following table lists the API operations in a logical order, grouped by resource type.

For information about permissions, see Permissions.

API Operation	Permissions Required to Use the Operation
`ListZones`	DNS_ZONE_INSPECT
`CreateZone`	DNS_ZONE_CREATE
`DeleteZone`	DNS_ZONE_DELETE
`GetZone`	DNS_ZONE_READ
`UpdateZone`	DNS_ZONE_UPDATE
`GetZoneRecords`	DNS_ZONE_READ and DNS_RECORD_READ
`PatchZoneRecords`	DNS_ZONE_UPDATE and DNS_RECORD_UPDATE
`UpdateZoneRecords`	DNS_ZONE_UPDATE and DNS_RECORD_UPDATE
`GetDomainRecords`	DNS_RECORD_READ
`PatchDomainRecords`	DNS_RECORD_UPDATE
`UpdateDomainRecords`	DNS_RECORD_UPDATE
`DeleteRRSet`	DNS_RECORD_UPDATE
`GetRRSet`	DNS_RECORD_READ
`PatchRRSet`	DNS_RECORD_UPDATE
`UpdateRRSet`	DNS_RECORD_UPDATE
`GetDNSTrafficCounts`	DNS_TRAFFIC_READ

API Operation	Permissions Required to Use the Operation
`ListSteeringPolicies`	DNS_STEERING_ POLICY_INSPECT
`CreateSteeringPolicy`	DNS_STEERING_ POLICY_CREATE
`GetSteeringPolicy`	DNS_STEERING_ POLICY_READ
`UpdateSteeringPolicy`	DNS_STEERING_ POLICY_UPDATE
`DeleteSteeringPolicy`	DNS_STEERING_ POLICY_DELETE
`ListSteeringPolicyAttachments`	DNS_STEERING_ ATTACHMENT_INSPECT
`CreateSteeringPolicyAttachment`	DNS_ZONE_UPDATE and DNS_STEERING_ POLICY_READ
`GetSteeringPolicyAttachment`	DNS_STEERING_ ATTACHMENT_READ
`UpdateSteeringPolicyAttachment`	DNS_ZONE_UPDATE and DNS_STEERING_ POLICY_READ
`DeleteSteeringPolicyAttachment`	DNS_ZONE_UPDATE and DNS_STEERING_ POLICY_READ

Details for the Email Service

This topic covers details for writing policies to control access to the Email service.

Resource-Types

`approved-senders`

`suppressions`

Supported Variables

The Email Service supports all the general variables (see General Variables for All Requests), plus the ones listed here.

The `approved-senders` resource type can use the following variables:

Variable	Variable Type	Comments
`target.approved-sender.id`	Entity (OCID)	
`target.approved-sender.emailaddress`	String	Use this variable with the APPROVED_SENDER_USE permissions only.

Details for Verb + Resource-Type Combinations

The following tables show the permissions and API operations covered by each verb. The level of access is cumulative as you go from `inspect` > `read` > `use` > `manage`. A plus sign (+) in a table cell indicates incremental access compared to the cell directly above it, whereas "no extra" indicates no incremental access.

For example, the `use` verb for the `suppressions` resource-type covers no extra permissions or API operations compared to the `read` verb.

approved-senders

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
APPROVED_SENDER_INSPECT	`ListSenders`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT + APPROVED_SENDER_READ	`GetSender`	*None*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + APPROVED_SENDER_USE	`SmtpSend`	*None*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + APPROVED_SENDER_CREATE APPROVED_SENDER_DELETE APPROVED_SENDER_UPDATE	`CreateSender` `DeleteSender` `UpdateSender`	*none*

suppressions

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
SUPPRESSION_INSPECT	`ListSuppression`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT + SUPPRESSION_READ	`GetSuppression`	*None*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
No extra	*None*	*None*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + SUPPRESSION_CREATE SUPPRESSION_DELETE	`CreateSuppression` `DeleteSuppression`	*none*

Permissions Required for Each API Operation

The following table lists the API operations in a logical order, grouped by resource type.

For information about permissions, see Permissions.

API Operation	Permissions Required to Use the Operation
`ListSenders`	APPROVED_SENDER_INSPECT
`GetSender`	APPROVED_SENDER_READ
`CreateSender`	APPROVED_SENDER_CREATE
`DeleteSender`	APPROVED_SENDER_DELETE
`SmtpSend`	APPROVED_SENDER_USE
`ListSuppression`	SUPPRESSION_INSPECT
`GetSuppression`	SUPPRESSION_READ
`CreateSuppression`	SUPPRESSION_CREATE
`DeleteSuppression`	SUPPRESSION_DELETE

Details for the File Storage Service

This topic covers details for writing policies to control access to the File Storage Service.

Aggregate Resource-Type

- `file-family`

Individual Resource-Types

- `file-systems`
- `mount-targets`
- `export-sets`

COMMENTS

A policy that uses `<verb> file-family` is equivalent to writing one with a separate `<verb> <individual resource-type>` statement for each of the individual resource-types.

See the table in Details for Verb + Resource-Type Combinations for a detailed breakout of the API operations covered by each verb, for each individual resource-type included in `file-family`.

Supported Variables

Only the general variables are supported (see General Variables for All Requests).

Details for Verb + Resource-Type Combinations

The following tables show the permissions and API operations covered by each verb. The level of access is cumulative as you go from `inspect` > `read` > `use` > `manage`. A plus sign (+) in a table cell indicates incremental access compared to the cell directly above it, whereas "no extra" indicates no incremental access..

For example, the `read` verb for the `file-systems` resource-type includes the same permissions and API operations as the `inspect` verb, plus the FILE_SYSTEM_READ permission and a number of API operations (e.g., `GetFileSystem`, `ListMountTargets`, etc.). The `use`

verb covers still another permission and set of API operations compared to `read`. Lastly, `manage` covers two more permissions and operations compared to `use`.

export-sets

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
EXPORT_SET_INSPECT	`ListExportSets`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT +	*INSPECT* +	*none*
EXPORT_SET_READ	`GetExport`	
	`GetExportSet`	
	`ListExports`	

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	*USE* +	`CreateExport`
EXPORT_SET_CREATE	`CreateExportSet`	`DeleteExport`
EXPORT_SET_UPDATE	`UpdateExportSet`	(both also need `use file-systems`.)
EXPORT_SET_DELETE	`DeleteExportSet`	

file-systems

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
FILE_SYSTEM_INSPECT	`ListFileSystems`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT +	*INSPECT* +	*none*
FILE_SYSTEM_READ	`GetFileSystem`	
	`GetSnapshot`	
	`ListSnapshots`	

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ +	*no extra*	`CreateExport`
FILE_SYSTEM_NFSv3_EXPORT		`DeleteExport`
FILE_SYSTEM_NFSv3_UNEXPORT		(both also need `manage export-sets`.)

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	*USE* +	*none*
FILE_SYSTEM_CREATE	`CreateFileSystem`	
FILE_SYSTEM_UPDATE	`UpdateFileSystem`	
FILE_SYSTEM_DELETE	`DeleteFileSystem`	
FILE_SYSTEM_CREATE_SNAPSHOT	`CreateSnapshot`	
FILE_SYSTEM_DELETE_SNAPSHOT	`DeleteSnapshot`	

mount-targets

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
MOUNT_TARGET_INSPECT	`ListMountTargets`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT +	*INSPECT* +	*none*
MOUNT_TARGET_READ	`GetMountTarget`	

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + MOUNT_TARGET_CREATE MOUNT_TARGET_UPDATE MOUNT_TARGET_DELETE	`UpdateMountTarget`	*USE* + `CreateMountTarget`, `DeleteMountTarget` (both also need `use vnics`, `use private-ips`, and `use subnets`.)

Permissions Required for Each API Operation

The following table lists the API operations in a logical order, grouped by resource type.



Tip

If a group uses the Console to create file systems, permissions to read mount targets is required. See the file storage policy examples for further guidance.

For information about permissions, see Permissions.

API Operation	Permissions Required to Use the Operation
`ListExports`	EXPORT_SET_READ
`CreateExport`	EXPORT_SET_UPDATE + FILE_SYSTEM_NFSv3_EXPORT
`GetExport`	EXPORT_SET_READ

API Operation	Permissions Required to Use the Operation
`DeleteExport`	EXPORT_SET_UPDATE + FILE_SYSTEM_NFSv3_UNEXPORT
`ListExportSets`	EXPORT_SET_INSPECT
`CreateExportSet`	EXPORT_SET_CREATE
`GetExportSet`	EXPORT_SET_READ
`UpdateExportSet`	EXPORT_SET_UPDATE
`DeleteExportSet`	EXPORT_SET_DELETE
`ListFileSystems`	FILE_SYSTEM_INSPECT
`CreateFileSystem`	FILE_SYSTEM_CREATE
`GetFileSystem`	FILE_SYSTEM_READ
`UpdateFileSystem`	FILE_SYSTEM_UPDATE
`DeleteFileSystem`	FILE_SYSTEM_DELETE
`ListMountTargets`	MOUNT_TARGET_INSPECT
`CreateMountTarget`	MOUNT_TARGET_CREATE + VNIC_CREATE(vnicCompartment) + SUBNET_ATTACH(subnetCompartment) + VNIC_ATTACH(vnicCompartment) + PRIVATE _IP_CREATE(subnetCompartment) + PRIVATE_IP_ASSIGN(subnetCompartment) + VNIC_ASSIGN(subnetCompartment)

API Operation	Permissions Required to Use the Operation
`GetMountTarget`	MOUNT_TARGET_READ
`UpdateMountTarget`	MOUNT_TARGET_UPDATE
`DeleteMountTarget`	MOUNT_TARGET_DELETE + VNIC_DELETE(vnicCompartment) + SUBNET_DETACH(subnetCompartment) + VNIC_DETACH(vnicCompartment) + PRIVATE_IP_DELETE(subnetCompartment) + PRIVATE_IP_UNASSIGN(subnetCompartment) + VNIC_UNASSIGN(vnicCompartment)
`ListSnapshots`	FILE_SYSTEM_READ
`CreateSnapshot`	FILE_SYSTEM_CREATE_SNAPSHOT
`GetSnapshot`	FILE_SYSTEM_READ
`DeleteSnapshot`	FILE_SYSTEM_DELETE_SNAPSHOT

Details for the Health Checks Service

This topic covers details for writing policies to control access to the Health Checks service.

Resource-Types

`health-check-monitor`

`health-check-results`

`on-demand-probe`

`vantage-points`

```
health-check-family
```

Supported Variables

The Health Checks Service supports all the general variables (see General Variables for All Requests), plus the ones listed here. Values in the list can be any valid test type. For example, HTTP, HTTPS, ICMP, etc.

Variable	Variable Type	Comments
`target.health-check-monitor.test-type`	String	
`target.on-demand-probe.test-type`	String	
`target.health-check-results.test-type`	String	

Details for Verb + Resource-Type Combinations

The following tables show the permissions and API operations covered by each verb. The level of access is cumulative as you go from `inspect` > `read` > `use` > `manage`. A plus sign (+) in a table cell indicates incremental access compared to the cell directly above it, whereas "no extra" indicates no incremental access.

For example, the `use` verb for the `health-check-monitor` resource-type covers no extra permissions or API operations compared to the `read` verb.

health-check-monitor

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
HEALTH_CHECK_MONITOR_INSPECT	`ListOHCMonitors`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT + HEALTH_CHECK_MONITOR_READ	`GetOHCMonitor`	*None*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
No extra	*None*	*None*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + HEALTH_CHECK_MONITOR_MANAGE	`CreateOHCMonitor` `UpdateOHCMonitor` `DeleteOHCMonitor`	*None*

health-check-results

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
No extra	*None*	*None*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
HEALTH_CHECK_RESULTS_READ	`ListOHCProbeResults` `ListOHCProbeResultsForTarget`	*None*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
No extra	*None*	*None*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
No extra	*None*	*None*

vantage-points

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
VANTAGE_POINTS_INSPECT	`ListVantagePoints`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
No extra	*None*	*None*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
No extra	*None*	*None*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
No extra	*None*	*None*

on-demand-probe

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
No extra	*None*	*None*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
No extra	*None*	*None*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
No extra	*None*	*None*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + ON_DEMAND_PROBE_MANAGE	`CreateOnDemandOHCProbe`	*None*

Permissions Required for Each API Operation

The following table lists the API operations in a logical order, grouped by resource type.

For information about permissions, see Permissions.

API Operation	Permissions Required to Use the Operation
`ListOHCMonitors`	HEALTH_CHECK_MONITOR_INSPECT
`CreateOHCMonitor`	HEALTH_CHECK_MONITOR_MANAGE
`GetOHCMonitor`	HEALTH_CHECK_MONITOR_READ
`UpdateOHCMonitor`	HEALTH_CHECK_MONITOR_MANAGE
`DeleteOHCMonitor`	HEALTH_CHECK_MONITOR_MANAGE
`ListOHCProbeResults`	HEALTH_CHECK_RESULTS_READ
`ListOHCProbeResultsForTarget`	HEALTH_CHECK_RESULTS_READ
`ListVantagePoints`	VANTAGE_POINTS_INSPECT
`CreateOnDemandOHCProbe`	ON_DEMAND_PROBE_MANAGE

Details for IAM

This topic covers details for writing policies to control access to IAM.

Resource-Types

```
authentication-policies
compartments
users
groups
dynamic-groups
policies
identity-providers
tenancies
tag-namespaces
tagdefinitions
tag-defaults
workrequest
```

Supported Variables

IAM supports all the general variables (see General Variables for All Requests), plus additional ones listed here:

Operations for This Resource-Type...	Can Use These Variables...	Variable Type	Comments
users	target.user.id	Entity (OCID)	Not available to use with CreateUser.
	target.user.name	String	
groups	target.group.id	Entity (OCID)	Not available to use with CreateGroup.
	target.group.name	String	
	target.group.member	Boolean	True if request.user is a member of target.group.
policies	target.policy.id	Entity (OCID)	Not available to use with CreatePolicy.
	target.policy.name	String	
	target.policy.autoupdate	Boolean	Whether the policy being acted upon uses "Keep policy current" as its version date (i.e., either null or an empty string for the versionDate parameter in CreatePolicy and UpdatePolicy).

Operations for This Resource-Type...	Can Use These Variables...	Variable Type	Comments
compartments	target.compartment.id	Entity (OCID)	For CreateCompartment, this will be the value of the parent compartment (e.g., the root compartment). This is a universal variable available to use with any request across all services (see General Variables for All Requests).
	target.compartment.name	String	
tag-namespace	target.tag-namespace.id	Entity (OCID)	This variable is supported only in statements granting permissions for the tag-namespaces resource-type. For an example, see Required Permissions for Working with Defined Tags. Not available to use with CreateTagNamespace.
	target.tag-namespace.name	String	

Details for Verbs + Resource-Type Combinations

The following tables show the permissions and API operations covered by each verb. The level of access is cumulative as you go from inspect > read > use > manage. A plus sign (+) in a table cell indicates incremental access compared to the cell directly above it, whereas "no extra" indicates no incremental access.

For example, the read verb for compartments covers no extra permissions or API operations compared to the inspect verb. The use verb includes the same ones as the read verb, plus

the COMPARTMENT_UPDATE permission and `UpdateCompartment` API operation. The `manage` verb includes the same permissions and API operations as the `use` verb, plus the COMPARTMENT_CREATE permission and two API operations: `CreateCompartment` and `DeleteCompartment`

authentication-policies

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
AUTHENTICATION_POLICY_INSPECT	`GetAuthenticationPolicy`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + AUTHENTICATION_POLICY_UPDATE	*USE* + `UpdateAuthenticationPolicy`	*none*

compartments

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
COMPARTMENT_INSPECT	`ListCompartments` `GetCompartment` `ListAvailabilityDomains` `ListFaultDomains`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ +	*READ* +	*none*
COMPARTMENT_UPDATE	`UpdateCompartment`	
	`GetWorkRequest`	

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	*USE* +	*none*
COMPARTMENT_CREATE	`CreateCompartment`	
COMPARTMENT_DELETE	`DeleteCompartment`	

users

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
USER_INSPECT	`ListUsers`	`GetUserGroupMembership` (also need `inspect groups`)
	`GetUser`	

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT +	*INSPECT* +	*no extra*
USER_READ	`ListApiKeys`	
	`ListSwiftPasswords`	
	`ListAuthTokens`	
	`ListCustomerSecretKeys`	
	`ListMfaTotpDevices`	

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ +	*READ* +	*READ* +
USER_UPDATE	`UpdateUser`	`AddUserToGroup` (also need `use groups`) `RemoveUserFromGroup` (also need `use groups`)

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	*USE* +	*no extra*
USER_CREATE	`CreateUser`	
USER_DELETE	`DeleteUser`	
USER_UNBLOCK	`UpdateUserState`	
USER_APIKEY_ADD	`UploadApiKey`	
USER_APIKEY_REMOVE	`DeleteApiKey`	
USER_UIPASS_SET	`CreateOrResetUIPassword`	
USER_UIPASS_RESET	`UpdateSwiftPassword`	
USER_SWIFTPASS_SET	`CreateSwiftPassword`	
USER_SWIFTPASS_RESET	`DeleteSwiftPassword`	
USER_SWIFTPASS_REMOVE	`UpdateAuthToken`	
USER_AUTHTOKEN_SET	`CreateAuthToken`	
USER_AUTHTOKEN_RESET	`DeleteAuthToken`	
USER_AUTHTOKEN_REMOVE	`CreateSecretKey`	
USER_SECRETKEY_ADD	`UpdateCustomerSecretKey`	
USER_SECRETKEY_UPDATE	`DeleteCustomerSecretKey`	
USER_SECRETKEY_REMOVE	`CreateMfaTotpDevice`	
USER_TOTPDEVICE_ADD	`ActivateMfaTotpDevice`	
USER_TOTPDEVICE_REMOVE	`DeleteMfaTotpDevice`	
USER_TOTPDEVICE_UPDATE		

groups

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
GROUP_INSPECT	`ListGroups` `GetGroup`	`GetUserGroupMembership` (also need `inspect users`) `ListIdpGroupMappings`, `GetIdpGroupMapping` (both also need `inspect identity-providers`)

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*no extra*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + GROUP_UPDATE	*READ* + `UpdateGroup`	*READ* + `AddUserToGroup` (also need `use users`) `RemoveUserFromGroup` (also need `use users`) `AddIdpGroupMapping`, `DeleteIdpGroupMapping` (both also need `manage identity-providers`)

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + GROUP_CREATE GROUP_DELETE	*USE* + `CreateGroup` `DeleteGroup`	*no extra*

dynamic-groups

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
DYNAMIC_GROUP_INSPECT	`ListDynamicGroups` `GetDynamicGroup`	*No extra*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*no extra*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + DYNAMIC_GROUP_UPDATE	*READ* + `UpdateDynamicGroup`	*No extra*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + DYNAMIC_GROUP_CREATE DYNAMIC_GROUP_DELETE	*USE* + `CreateDynamicGroup` `DeleteDynamicGroup`	*no extra*

policies

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
POLICY_READ	`ListPolicies` `GetPolicy`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra* **Note:** The ability to update policies is available only with `manage policies`.	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + POLICY_UPDATE POLICY_CREATE POLICY_DELETE	*USE* + `UpdatePolicy` `CreatePolicy` `DeletePolicy`	*none*

identity-providers

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
IDENTITY_PROVIDER_INSPECT	`ListIdentityProviders` `GetIdentityProvider`	`ListIdpGroupMappings`, `GetIdpGroupMapping` (both also need `inspect groups`)

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*no extra*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*no extra*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	*USE* +	*USE* +
IDENTITY_PROVIDER_UPDATE	`UpdateIdentityProvider`	`AddIdpGroupMapping`,
IDENTITY_PROVIDER_CREATE	`CreateIdentityProvider`	`DeleteIdpGroupMapping` (both also need `use groups`)
IDENTITY_PROVIDER_DELETE	`DeleteIdentityProvider`	

tenancies

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
TENANCY_INSPECT	`ListRegionSubscriptions`	*none*
	`GetTenancy`	
	`ListRegions`	

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ +	*no extra*	*none*
TENANCY_UPDATE		

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	*USE* +	*none*
TENANCY_UPDATE	`CreateRegionSubscription`	

tag-namespaces

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
TAG_NAMESPACE_INSPECT	`ListTagNamespaces` `GetTagNamespace`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + TAG_NAMESPACE_USE	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + TAG_NAMESPACE_UPDATE TAG_NAMESPACE_CREATE TAG_NAMESPACE_MOVE	*USE* + `UpdateTagNamespace` `CreateTagNamespace` `ChangeTagNamespaceCompartment`	*none*

tagdefinitions

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
TAG_DEFINITION_INSPECT	`ListTagDefinitions` `GetTagDefinition`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + TAG_DEFINITION_UPDATE	*READ* + `UpdateTagDefinition`	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + TAG_DEFINITION_CREATE	*USE* + `CreateTagDefinition`	*none*

tag-defaults

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
TAG_DEFAULT_INSPECT	`ListTagDefaults` `GetTagDefault`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE		
Permissions	**APIs Fully Covered**	**APIs Partially Covered**
INSPECT +	*USE +*	*none*
TAG_DEFAULT_CREATE	`CreateTagDefault`	
TAG_DEFAULT_UPDATE	`UpdateTagDefault`	
TAG_DEFAULT_DELETE	`DeleteTagDefault`	

Permissions Required for Each API Operation

The following table lists the API operations in a logical order, grouped by resource type.

For information about permissions, see Permissions.

API Operation	Permissions Required to Use the Operation
`ListRegions`	TENANCY_INSPECT
`ListRegionSubscriptions`	TENANCY_INSPECT
`CreateRegionSubscription`	TENANCY_UPDATE
`GetTenancy`	TENANCY_INSPECT
`GetAuthenticationPolicy`	AUTHENTICATION_POLICY_INSPECT
`UpdateAuthenticationPolicy`	AUTHENTICATION_POLICY_UPDATE
`ListAvailabilityDomains`	COMPARTMENT_INSPECT
`ListFaultDomains`	COMPARTMENT_INSPECT
`ListCompartments`	COMPARTMENT_INSPECT
`GetCompartment`	COMPARTMENT_INSPECT
`UpdateCompartment`	COMPARTMENT_UPDATE

API Operation	Permissions Required to Use the Operation
CreateCompartment	COMPARTMENT_CREATE
DeleteCompartment	COMPARTMENT_DELETE
GetWorkRequest	COMPARTMENT_READ
ListUsers	USER_INSPECT
GetUser	USER_INSPECT
UpdateUser	USER_UPDATE
UpdateUserState	USER_UPDATE and USER_UNBLOCK
CreateUser	USER_CREATE
DeleteUser	USER_DELETE
CreateOrResetUIPassword	USER_UPDATE and USER_UIPASS_RESET
ListApiKeys	USER_READ
UploadApiKey	USER_UPDATE and USER_APIKEY_ADD
DeleteApiKey	USER_UPDATE and USER_APIKEY_REMOVE
ListAuthTokens	USER_READ
UpdateAuthToken	USER_UPDATE and USER_AUTHTOKEN_RESET
CreateAuthToken	USER_UPDATE and USER_AUTHTOKEN_SET
DeleteAuthToken	USER_UPDATE and USER_AUTHTOKEN_REMOVE
ListSwiftPasswords	USER_READ
UpdateSwiftPassword	USER_UPDATE and USER_SWIFTPASS_RESET

API Operation	Permissions Required to Use the Operation
CreateSwiftPassword	USER_UPDATE and USER_SWIFTPASS_SET
DeleteSwiftPassword	USER_UPDATE and USER_SWIFTPASS_REMOVE
ListCustomerSecretKeys	USER_READ
CreateSecretKey	USER_UPDATE and USER_SECRETKEY_ADD
UpdateCustomerSecretKey	USER_UPDATE and USER_SECRETKEY_UPDATE
DeleteCustomerSecretKey	USER_UPDATE and USER_SECRETKEY_REMOVE
ListUserGroupMemberships	GROUP_INSPECT and USER_INSPECT
GetUserGroupMembership	USER_INSPECT and GROUP_INSPECT
AddUserToGroup	GROUP_UPDATE and USER_UPDATE
RemoveUserFromGroup	GROUP_UPDATE and USER_UPDATE
ListGroups	GROUP_INSPECT
GetGroup	GROUP_INSPECT
UpdateGroup	GROUP_UPDATE
CreateGroup	GROUP_CREATE
DeleteGroup	GROUP_DELETE
ListDynamicGroups	DYNAMIC_GROUP_INSPECT
GetDynamicGroup	DYNAMIC_GROUP_INSPECT
UpdateDynamicGroup	DYNAMIC_GROUP_UPDATE
CreateDynamicGroup	DYNAMIC_GROUP_CREATE

API Operation	Permissions Required to Use the Operation
DeleteDynamicGroup	DYNAMIC_GROUP_DELETE
ListPolicies	POLICY_READ
GetPolicy	POLICY_READ
UpdatePolicy	POLICY_UPDATE
CreatePolicy	POLICY_CREATE
DeletePolicy	POLICY_DELETE
ListIdentityProviders	IDENTITY_PROVIDER_INSPECT
GetIdentityProvider	IDENTITY_PROVIDER_INSPECT
UpdateIdentityProvider	IDENTITY_PROVIDER_UPDATE
CreateIdentityProvider	IDENTITY_PROVIDER_CREATE
DeleteIdentityProvider	IDENTITY_PROVIDER_DELETE
ListIdpGroupMappings	IDENTITY_PROVIDER_INSPECT and GROUP_INSPECT
GetIdpGroupMapping	IDENTITY_PROVIDER_INSPECT and GROUP_INSPECT
AddIdpGroupMapping	IDENTITY_PROVIDER_UPDATE and GROUP_UPDATE
DeleteIdpGroupMapping	IDENTITY_PROVIDER_UPDATE and GROUP_UPDATE
ListTagNamespaces	TAG_NAMESPACE_INSPECT
GetTagNamespace	TAG_NAMESPACE_INSPECT
CreateTagNamespace	TAG_NAMESPACE_CREATE
UpdateTagNamespace	TAG_NAMESPACE_UPDATE

API Operation	Permissions Required to Use the Operation
`ChangeTagNamespaceCompartment`	TAG_NAMESPACE_MOVE
`ListTagDefinitions`	TAG_DEFINITION_INSPECT
`GetTagDefinition`	TAG_DEFINITION_INSPECT
`CreateTagDefinition`	TAG_DEFINITION_CREATE
`UpdateTagDefinition`	TAG_DEFINITION_UPDATE
`ListTagDefaults`	TAG_DEFAULT_INSPECT
`GetTagDefault`	TAG_DEFAULT_INSPECT
`CreateTagDefault`	TAG_DEFAULT_MANAGE
`UpdateTagDefault`	TAG_DEFAULT_MANAGE
`DeleteTagDefault`	TAG_DEFAULT_MANAGE

Details for the Key Management Service

This topic covers details for writing policies to control access to the Key Management service.

Resource-Types

`vaults`

`keys`

`key-delegates`

Supported Variables

Key Management supports all the general variables, plus the ones listed here. For more information about general variables supported by Oracle Cloud Infrastructure services, see

General Variables for All Requests.

Variable	Variable Type	Comments
`target.key.id`	Entity (OCID)	Use this variable to control access to specific keys by OCID.
`target.vault.id`	Entity (OCID)	Use this variable to control access to specific vaults by OCID.
`request.includePlainTextKey`	String	Use this variable to control whether to return the plaintext key in addition to the encrypted key in response to a request to generate a data encryption key.
`request.kms-key.id`	String	Use this variable to control whether block volumes or buckets can be created without a Key Management master encryption key.
`target.boot-volume.kms-key.id`	String	Use this variable to control whether Compute instances can be launched with boot volumes that were created without a Key Management master encryption key.

Details for Verb + Resource-Type Combinations

The following tables show the permissions and API operations covered by each verb. The level of access is cumulative as you go from `inspect` > `read` > `use` > `manage`. A plus sign (+) in a table cell indicates incremental access compared to the cell directly above it, whereas "no extra" indicates no incremental access.

For example, the `use` verb for the `keys` resource-type includes the same permissions and API operations as the `read` verb, plus the KEY_ENCRYPT and KEY_DECRYPT permissions and a number of API operations (`Encrypt`, `Decrypt`, and `GenerateDataEncryptionKey`). The

`manage` verb allows even more permissions and API operations when compared to the `use` verb.

vaults

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
VAULT_INSPECT	`ListVaults`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT + VAULT_READ	*INSPECT* + `GetVault`	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + VAULT_CREATE_KEY	*READ* + `CreateKey`	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + VAULT_CREATE VAULT_UPDATE VAULT_DELETE	*USE* + `CreateVault` `UpdateVault` `ScheduleVaultDeletion` `CancelVaultDeletion`	*none*

keys

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
KEY_INSPECT	`ListKeys` `ListKeyVersions`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT + KEY_READ	*INSPECT* + `GetKey` `GetKeyVersion`	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + KEY_ENCRYPT KEY_DECRYPT	*READ* + `Encrypt` `GenerateDataEncryptionKey` `Decrypt`	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + KEY_CREATE KEY_UPDATE KEY_ROTATE	*USE* + `CreateKey` `UpdateKey` `DisableKey` `EnableKey` `CreateKeyVersion`	*none*

key-delegate

USE

Permissions	APIs Fully Covered	APIs Partially Covered
KEY_ASSOCIATE	`Encrypt`	*none*
KEY_DISASSOCIATE	`GenerateDataEncryptionKey`	
	`Decrypt`	

Permissions Required for Each API Operation

The following table lists the API operations in a logical order, grouped by resource type.

For information about permissions, see Permissions.

API Operation	Permissions Required to Use the Operation
`ListVaults`	VAULT_INSPECT
`GetVault`	VAULT_READ
`CreateVault`	VAULT_CREATE
`UpdateVault`	VAULT_UPDATE
`ScheduleVaultDeletion`	VAULT_DELETE
`CancelVaultDeletion`	VAULT_DELETE
`ListKeys`	KEY_INSPECT
`ListKeyVersions`	KEY_INSPECT
`GetKey`	KEY_READ
`CreateKey`	KEY_CREATE and VAULT_CREATE_KEY
`EnableKey`	KEY_UPDATE

API Operation	Permissions Required to Use the Operation
`DisableKey`	KEY_UPDATE
`UpdateKey`	KEY_UPDATE
`GetKeyVersion`	KEY_READ
`CreateKeyVersion`	KEY_ROTATE
`GenerateDataEncryptionKey`	KEY_ENCRYPT
`Encrypt`	KEY_ENCRYPT
`Decrypt`	KEY_DECRYPT

Details for Load Balancing

This topic covers details for writing policies to control access to the Load Balancing service.

Resource-Types

`load-balancers`

Supported Variables

Only the general variables are supported (see General Variables for All Requests).

Details for Verb + Resource-Type Combinations

The following tables show the permissions and API operations covered by each verb. The level of access is cumulative as you go from `inspect` > `read` > `use` > `manage`. A plus sign (+) in a table cell indicates incremental access compared to the cell directly above it, whereas "no extra" indicates no incremental access..

For example, the `read` verb for load-balancers includes the same permissions and API operations as the `inspect` verb, plus the LOAD_BALANCER_READ permission and a number of

API operations (e.g., `GetLoadBalancer`, `ListWorkRequests`, etc.). The `use` verb covers still another permission and set of API operations compared to `read`. And `manage` covers two more permissions and operations compared to `use`.

LOAD-BALANCERS

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
LOAD_BALANCER_INSPECT	`ListLoadBalancers`	*none*
	`ListShapes`	
	`ListPolicies`	
	`ListProtocols`	

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT +	*INSPECT +*	*none*
LOAD_BALANCER_READ	`GetLoadBalancer`	
	`ListWorkRequests`	
	`GetWorkRequest`	
	`ListBackendSets`	
	`GetBackendSet`	
	`ListBackends`	
	`GetBackend`	
	`GetHealthChecker`	
	`ListCertificates`	

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ +	*READ* +	*none*
LOAD_BALANCER_UPDATE	`UpdateLoadBalancer`	
	`UpdateBackendSet`	
	`CreateBackendSet`	
	`DeleteBackendSet`	
	`UpdateBackend`	
	`CreateBackend`	
	`DeleteBackend`	
	`UpdateHealthChecker`	
	`CreateCertificate`	
	`DeleteCertificate`	
	`UpdateListener`	
	`CreateListener`	
	`DeleteListener`	

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	*USE* +	*none*
LOAD_BALANCER_CREATE	`CreateLoadBalancer`	
LOAD_BALANCER_DELETE	`DeleteLoadBalancer`	

Permissions Required for Each API Operation

The following table lists the API operations in a logical order, grouped by resource type.



Tip

If a group uses the Console to manage load balancers, permissions to use the associated networking resources are required. See the load balancing policy examples for further guidance.

For information about permissions, see Permissions.

API Operation	Permissions Required to Use the Operation
`ListLoadBalancers`	LOAD_BALANCER_INSPECT and
`GetLoadBalancer`	LOAD_BALANCER_READ
`UpdateLoadBalancer`	LOAD_BALANCER_UPDATE
`CreateLoadBalancer`	LOAD_BALANCER_CREATE
`DeleteLoadBalancer`	LOAD_BALANCER_DELETE
`ListShapes`	LOAD_BALANCER_INSPECT
`ListWorkRequests`	LOAD_BALANCER_READ
`GetWorkRequest`	LOAD_BALANCER_READ
`ListBackendSets`	LOAD_BALANCER_READ
`GetBackendSet`	LOAD_BALANCER_READ
`UpdateBackendSet`	LOAD_BALANCER_UPDATE
`CreateBackendSet`	LOAD_BALANCER_UPDATE
`DeleteBackendSet`	LOAD_BALANCER_UPDATE
`ListBackends`	LOAD_BALANCER_READ
`GetBackend`	LOAD_BALANCER_READ
`UpdateBackend`	LOAD_BALANCER_UPDATE
`CreateBackend`	LOAD_BALANCER_UPDATE
`DeleteBackend`	LOAD_BALANCER_UPDATE
`GetHealthChecker`	LOAD_BALANCER_READ

API Operation	Permissions Required to Use the Operation
`UpdateHealthChecker`	LOAD_BALANCER_UPDATE
`ListCertificates`	LOAD_BALANCER_READ
`CreateCertificate`	LOAD_BALANCER_UPDATE
`DeleteCertificate`	LOAD_BALANCER_UPDATE
`UpdateListener`	LOAD_BALANCER_UPDATE
`CreateListener`	LOAD_BALANCER_UPDATE
`DeleteListener`	LOAD_BALANCER_UPDATE
`ListPolicies`	LOAD_BALANCER_INSPECT
`ListProtocols`	LOAD_BALANCER_INSPECT

Details for Monitoring

This topic covers details for writing policies to control access to the Monitoring service.

Resource-Types

`alarms`

`metrics`

Supported Variables

Monitoring supports all the general variables (see General Variables for All Requests), plus the one listed here:

Operations for This Resource-Type...	Can Use This Variable	Variable Type	Comments
`metrics`	`target.metrics.namespace`	String	Use this variable to control access to specific resource types. Surround the namspace value with single quotes. For example, to control access to metrics for Compute instances, use the following phrase: `where target.metrics.namespace='oci_computeagent'` For an example policy, see Restrict user access to a specific metric namespace. For valid namespace values, see Supported Services.

Details for Verb + Resource-Type Combinations

The following tables show the permissions and API operations covered by each verb. The level of access is cumulative as you go from `inspect` > `read` > `use` > `manage`. A plus sign (+) in a table cell indicates incremental access compared to the cell directly above it, whereas "no extra" indicates no incremental access.

alarms

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
ALARM_INSPECT	`ListAlarms` `ListAlarmsStatus`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT + ALARM_READ	`GetAlarmHistory`	`GetAlarm` (also need `METRIC_READ` for the metric compartment and metric namespace)

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + *no extra*	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE + ALARM_CREATE ALARM_UPDATE ALARM_DELETE	`DeleteAlarm` `RemoveAlarmSuppression`	`CreateAlarm` (also need `METRIC_READ` for the metric compartment and metric namespace) `UpdateAlarm` (also need `METRIC_READ` for the metric compartment and metric namespace)

metrics

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
METRIC_INSPECT	`ListMetrics`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT + METRIC_READ	`SummarizeMetricsData`	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ + METRIC_WRITE	`PostMetricData`	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	*no extra*	*none*
no extra		

Permissions Required for Each API Operation

The following table lists the API operations in a logical order, grouped by resource type.

For information about permissions, see Permissions.

API Operation	Permissions Required to Use the Operation
`ListMetrics`	METRIC_INSPECT or METRIC_READ
`SummarizeMetricsData`	METRIC_READ
`PostMetricData`	METRIC_WRITE
`ListAlarms`	ALARM_INSPECT
`ListAlarmsStatus`	ALARM_INSPECT
`GetAlarm`	ALARM_READ and METRIC_READ
`GetAlarmHistory`	ALARM_READ
`CreateAlarm`	ALARM_CREATE and METRIC_READ
`UpdateAlarm`	ALARM_UPDATE and METRIC_READ
`RemoveAlarmSuppression`	ALARM_UPDATE
`DeleteAlarm`	ALARM_DELETE

Details for the Notifications Service

This topic covers details for writing policies to control access to the Notifications service.

Aggregate Resource-Type

`ons-family`

Individual Resource-Types

`ons-topics`

`ons-subscriptions`

Supported Variables

Only the general variables are supported (see General Variables for All Requests).

Details for Verb + Resource-Type Combinations

The following tables show the permissions and API operations covered by each verb. The level of access is cumulative as you go from `inspect` > `read` > `use` > `manage`. A plus sign (+) in a table cell indicates incremental access compared to the cell directly above it, whereas "no extra" indicates no incremental access..

ons-topics

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
ONS_TOPIC_INSPECT	`ListTopics`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT + ONS_TOPIC_READ	`GetTopic`	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ +	`PublishMessage`	*none*
ONS_TOPIC_PUBLISH		

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	`CreateTopic`	*none*
ONS_TOPIC_CREATE	`UpdateTopic`	
ONS_TOPIC_UPDATE	`DeleteTopic`	
ONS_TOPIC_DELETE		

ons-subscriptions

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
none	*none*	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT +	*no extra*	*none*
no extra		

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ +	*no extra*	*none*
no extra		

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	ListSubscriptions	*none*
ONS_TOPIC_SUBSCRIBE	CreateSubscription	
	UpdateSubscription	
	DeleteSubscription	
	GetSubscription	
	ResendSubscriptionConfirmation	

Permissions Required for Each API Operation

The following table lists the API operations in a logical order, grouped by resource type.

For information about permissions, see Permissions.

API Operation	Permissions Required to Use the Operation
ListTopics	ONS_TOPIC_INSPECT
GetTopic	ONS_TOPIC_READ
CreateTopic	ONS_TOPIC_CREATE
UpdateTopic	ONS_TOPIC_UPDATE
DeleteTopic	ONS_TOPIC_DELETE
ListSubscriptions	ONS_TOPIC_SUBSCRIBE
CreateSubscription	ONS_TOPIC_SUBSCRIBE
UpdateSubscription	ONS_TOPIC_SUBSCRIBE
DeleteSubscription	ONS_TOPIC_SUBSCRIBE
GetSubscription	ONS_TOPIC_SUBSCRIBE

API Operation	Permissions Required to Use the Operation
`GetConfirmSubscription`	(no permissions required; available to anyone)
`ResendSubscriptionConfirmation`	ONS_TOPIC_SUBSCRIBE
`GetUnsubscription`	(no permissions required; available to anyone)
`PublishMessage`	ONS_TOPIC_PUBLISH

Details for Object Storage, Archive Storage, and Data Transfer

This topic covers details for writing policies to control access to Archive Storage, Object Storage, and Data Transfer.



Tip

The object lifecycle policies feature requires that you grant permissions to the Object Storage service to archive and delete objects on your behalf. See Using Object Lifecycle Policies for more information.

Resource-Types

Individual Resource-Types

`objectstorage-namespaces`

`buckets`

`objects`

Aggregate Resource-Type

`object-family`

A policy that uses `<verb> object-family` is equivalent to writing one with a separate `<verb> <individual resource-type>` statement for each of the individual resource-types.

See the table in Details for Verb + Resource-Type Combinations for a detailed breakout of the API operations covered by each verb, for each individual resource-type included in `object-family`.

ADDITIONAL INDIVIDUAL RESOURCE-TYPE FOR DATA TRANSFER

`data-transfer-jobs`

Supported Variables

Object Storage supports all the general variables (see General Variables for All Requests), plus the ones listed here:

Operations for This Resource-Type...	Can Use This Variable	Variable Type	Comments
`buckets` and `objects`	`target.bucket.name`	String	Use this variable to control access to a specific bucket. For an example policy, see Let users write objects to Object Storage buckets.
`buckets` and `objects`	`request.vcn.id`	String	If you're using a service gateway with your VCN, you can use this variable to allow access to a bucket only from a specific virtual cloud network (VCN). For an example policy, see Task 4: (Optional) Update IAM Policies to Restrict Object Storage Bucket Access.

Operations for This Resource-Type...	Can Use This Variable	Variable Type	Comments
`buckets` and `objects`	`request.ipv4.ipaddress`	String	If you're using a service gateway with your VCN, you can use this variable to allow access to a bucket only from a specific CIDR range. For an example policy, see Task 4: (Optional) Update IAM Policies to Restrict Object Storage Bucket Access.

Details for Verb + Resource-Type Combinations

The following tables show the permissions and API operations covered by each verb. The level of access is cumulative as you go from `inspect` > `read` > `use` > `manage`. A plus sign (+) in a table cell indicates incremental access compared to the cell directly above it, whereas "no extra" indicates no incremental access.

FOR OBJECT-FAMILY RESOURCE TYPES

objectstorage-namespaces

READ

Permissions	APIs Fully Covered	APIs Partially Covered
None	`GetNamespace`	*none*
Permissions	**APIs Fully Covered**	
OBJECTSTORAGE_NAMESPACE_READ	`GetNamespace` with optional `compartmentId` parameter `GetNamespaceMetadata`	

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
READ +	*READ* +	*none*
OBJECTSTORAGE_NAMESPACE_UPDATE	`UpdateNamespaceMetadata`	

buckets

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
BUCKET_INSPECT	`HeadBucket`	*none*
	`ListBuckets`	

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT +	*INSPECT* +	*none*
BUCKET_READ	`GetBucket`	
	`ListMultipartUploads`	
	`GetObjectLifecyclePolicy`	

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ +	*READ* +	`PutObjectLifecyclePolicy`
BUCKET_UPDATE	`UpdateBucket`	
	`DeleteObjectLifecyclePolicy`	

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	*USE* +	*none*
BUCKET_CREATE	`CreateBucket`	
BUCKET_DELETE	`DeleteBucket`	
PAR_MANAGE	`CreatePar`	
	`GetPar`	
	`ListPar`	
	`DeletePar`	

objects

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
OBJECT_INSPECT	`HeadObject`	*none*
	`ListObjects`	
	`ListMultipartUploadParts`	

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT +	*INSPECT* +	*none*
OBJECT_READ	`GetObject`	

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ +	*READ* +	*READ* +
OBJECT_OVERWRITE	`PutObject`	`CreateMultipartUpload`, `UploadPart`, `CommitMultipartUpload` (these operations also need `manage objects`)

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	*USE* +	`PutObjectLifecyclePolicy` (also needs `manage objects`)
OBJECT_CREATE	`CreateObject`	
OBJECT_DELETE	`RenameObject`	
OBJECT_RESTORE	`RestoreObject`	
	`DeleteObject`	
	`CreateMultipartUpload`	
	`UploadPart`	
	`CommitMultipartUpload`	
	`AbortMultipartUpload`	

data-transfer-jobs

Policies for data transfer jobs also require either manage objects or manage objects and manage buckets. See Creating the Required IAM Users, Groups, and Policies for details.

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
TRANSFER_JOB_INSPECT	*no customer-facing API*	*no customer-facing API*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT +	*INSPECT* +	*no customer-facing API*
TRANSFER_JOB_READ	*no customer-facing API*	

USE

Permissions	APIs Fully Covered	APIs Partially Covered
READ +	*READ* +	*READ* +
TRANSFER_JOB_UPDATE	*no customer-facing API*	*no customer-facing API*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	*USE +*	*USE +*
TRANSFER_JOB_CREATE	*no customer-facing API*	*no customer-facing API*
TRANSFER_JOB_DELETE		

Permissions Required for Each API Operation

The following table lists the API operations in a logical order, grouped by resource type.

For information about permissions, see Permissions.

API Operation	Permissions Required to Use the Operation
`GetNamespace`	API requires no permissions and returns the caller's namespace. Use the API to validate your credentials. OBJECTSTORAGE_NAMESPACE_READ permission is required if you include the optional `compartmentId` parameter. Use the `compartmentId` parameter to determine the namespace for a third-party tenancy.
`GetNamespaceMetadata`	OBJECTSTORAGE_NAMESPACE_READ
`UpdateNamespaceMetadata`	OBJECTSTORAGE_NAMESPACE_UPDATE
`CreateBucket`	BUCKET_CREATE
`UpdateBucket`	BUCKET_UPDATE
`GetBucket`	BUCKET_READ
`HeadBucket`	BUCKET_INSPECT
`ListBuckets`	BUCKET_INSPECT
`DeleteBucket`	BUCKET_DELETE

API Operation	Permissions Required to Use the Operation
`PutObject`	The permission required depends on whether or not the object already exists in the bucket: • OBJECT_CREATE is required when an object with that name does not already exist in the bucket. • OBJECT_OVERWRITE is required when an object with that name already exists in the bucket.
`RenameObject`	OBJECT_CREATE and OBJECT_OVERWRITE
`GetObject`	OBJECT_READ
`HeadObject`	OBJECT_READ or OBJECT_INSPECT
`DeleteObject`	OBJECT_DELETE
`ListObjects`	OBJECT_INSPECT
`RestoreObjects`	OBJECT_RESTORE
`CreateMultipartUpload`	OBJECT_CREATE and OBJECT_OVERWRITE
`UploadPart`	OBJECT_CREATE and OBJECT_OVERWRITE
`CommitMultipartUpload`	OBJECT_CREATE and OBJECT_OVERWRITE
`ListMultipartUploadParts`	OBJECT_INSPECT
`ListMultipartUploads`	BUCKET_READ
`AbortMultipartUpload`	OBJECT_DELETE
`CreatePar`	PAR_MANAGE
`GetPar`	PAR_MANAGE

API Operation	Permissions Required to Use the Operation
`ListPars`	PAR_MANAGE
`DeletePar`	PAR_MANAGE
`PutObjectLifecyclePolicy`	BUCKET_UPDATE, OBJECT_CREATE, and OBJECT_DELETE
`GetObjectLifecyclePolicy`	BUCKET_READ
`DeleteObjectLifecyclePolicy`	BUCKET_UPDATE
`CreateCopyRequest`	OBJECT_READ, OBJECT_CREATE, OBJECT_OVERWRITE, and OBJECT_INSPECT
`GetWorkRequest`	OBJECT_READ
`ListWorkRequests`	OBJECT_INSPECT
`CancelWorkRequest`	OBJECT_DELETE

Details for Registry

This topic covers details for writing policies to control access to the Registry.

Resource-Types

- `repos`

Supported Variables

Oracle Cloud Infrastructure Registry supports all the general variables (see General Variables for All Requests), plus the ones listed here.

The `repos` resource-type can use the following variables:

Variable	Variable Type	Comments
`target.repo.name`	String	Use this variable to control access to specific repositories. For an example policy, see Policies to Control Repository Access.

Details for Verb + Resource-Type Combinations

The following tables show the permissions and API operations covered by each verb. The level of access is cumulative as you go from `inspect` > `read` > `use` > `manage`. A plus sign (+) in a table cell indicates incremental access compared to the cell directly above it, whereas "no extra" indicates no incremental access..

For example, the `read` verb for the `repos` resource-type includes the same permissions and API operations as the `inspect` verb, plus the REPOSITORY_READ permission and a number of API operations (e.g., `ReadDockerRepositoryMetadata`, etc.). The `use` verb covers still another permission and API operation compared to `read`. Lastly, `manage` covers more permissions and operations compared to `use`.

Note the Registry API is not currently available.

repos

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
REPOSITORY_INSPECT	`ListDockerRepositories` `ListDockerRepositoryManifests`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
INSPECT +	*INSPECT* +	*none*
REPOSITORY_READ	`ReadDockerRepositoryMetadata`	
	`ReadDockerRepositoryManifest`	
	`PullDockerLayer`	

USE

Permissions	APIs Fully Covered	APIs Partially Covered
no extra	*no extra*	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
USE +	*USE* +	*none*
REPOSITORY_CREATE	`CreateDockerRepository`	
REPOSITORY_DELETE	`DeleteDockerRepository`	
REPOSITORY_UPDATE	`UploadDockerImage`	
REPOSITORY_MANAGE	`DeleteDockerImage`	
	`DeleteDockerLayer`	
	`UpdateDockerRepositoryMetadata`	

Permissions Required for Each API Operation

The following table lists the API operations in a logical order, grouped by resource type.

Note the Registry API is not currently available.

For information about permissions, see Permissions.

API Operation	Permissions Required to Use the Operation
`ListDockerRepositories`	REPOSITORY_INSPECT
`ListDockerRepositoryManifests`	REPOSITORY_INSPECT

API Operation	Permissions Required to Use the Operation
`ReadDockerRepositoryMetadata`	REPOSITORY_READ
`ReadDockerRepositoryManifest`	REPOSITORY_READ
`CreateDockerRepository`	REPOSITORY_CREATE
`DeleteDockerRepository`	REPOSITORY_DELETE
`DeleteDockerRepositoryContents`	REPOSITORY_UPDATE
`UpdateDockerRepositoryMetadata`	REPOSITORY_MANAGE
`UploadDockerImage`	REPOSITORY_UPDATE + REPOSITORY_CREATE
`DeleteDockerImage`	REPOSITORY_UPDATE
`DeleteDockerLayer`	REPOSITORY_UPDATE
`PullDockerLayer`	REPOSITORY_READ
`UploadDockerLayer`	REPOSITORY_UPDATE + REPOSITORY_CREATE

Details for Resource Manager

This topic covers details for writing policies to control access to Resource Manager.

Resource Types

The Resource Manager supports two permission sets: one for stack resources and another for job resources.

Resource Type	Permissions
Stacks (`orm-stacks`)	inspect orm-stack read orm-stack use orm-stack create orm-stack update orm-stack delete orm-stack
Jobs (`orm-jobs`)	inspect orm-job read orm-job manage orm-job

Details for Verb + Resource-Type Combinations

Permissions, which are exposed as verbs in the Resource Manager policies, are implemented in a hierarchy. Following are the permission verbs with their policy scopes listed in ascending order, from least permissive to most permissive.

- **Inspect**: Allows you to call for a list of stacks or jobs, but not to act on the resources.
- **Read**: Allows you to get stacks and jobs, as well as Terraform configurations, state, logs, and execution plans, plus operations allowed by **Inspect**.
- **Use**: Allows you to create jobs, plus operations allowed by **Inspect** and **Read**.
- **Manage**: Includes all permissions, including permission to **create**, **update**, and **delete** stacks and jobs.



Note

The **use** verb is a special case. Only one operation API, `CreateJob`, uses it. Also, the operation requires two permissions, `JOB_MANAGE` and `STACK_USE`.

Verb	Stacks	Jobs
Inspect	inspect orm-stack	inspect orm-job
Read	inspect orm-stack read orm-stack	inspect orm-job read orm-job
Use	inspect orm-stack read orm-stack use orm-stack	inspect orm-job read orm-job
Manage	inspect orm-stack read orm-stack use orm-stack create orm-stack update orm-stack delete orm-stack	inspect orm-job read orm-job manage orm-job

Permissions Required for Each API Operation

The following table lists the Resource Manager API operations grouped by resource type, listed in alphabetical order.

For information about permissions, see Permissions.

Note that the create job operation requires two permissions, one for stack operations and one for job operations.

Operation (API)	Permission
List stacks (ListStacks)	inspect orm-stack
Create stack (CreateStack)	create orm-stack
Get stack (`GetStack`)	read orm-stack
Update stack (`UpdateStack`)	update orm-stack
Delete stack (`DeleteStack`)	delete orm-stack
Get stack Terraform configuration (`GetStackTfConfig`)	read orm-stack
List jobs (`ListJobs`)	inspect orm-job
Create job (`CreateJob`)	manage orm-job *and* use orm-stack
Get job (`GetJob`)	read orm-job
Update job (`UpdateJob`)	manage orm-job
Cancel job (`CancelJob`)	manage orm-job
Get job Terraform state file (`GetJobTfState`)	read orm-job
Get job Terraform configuration (`GetJobTfConfig`)	read orm-job
Get job Terraform execution plan (`GetJobTfExecutionPlan`)	read orm-job
Get job logs (`GetJobLogs`)	read orm-job

Details for Search

The Search service does not require permissions for its API operations. You do not need to write policies specifically to control access to Search. However, what you can see in search or query results depends on the permissions you have. If a policy exists to give you access to the `inspect` verb for a particular resource type, you have access to the permissions needed to view that resource type and its associated metadata in search results. If a service does not recognize the `inspect` verb or if the resource type's `inspect` verb does not fully cover list operations, permissions to view the service's supported resource types are granted by the `read` verb instead.

For more information about permissions, see the Permissions section of Advanced Policy Features.

Permissions Required to View Each Resource Type

The following table lists the resource types grouped by service, which are listed in alphabetical order. The Search API operations that can access the metadata for these resource types with these permissions are `GetResourceType`, `ListResourceTypes`, and `SearchResources`.

Service	Resource Type	Permissions Required to View in Search Results
Block Volume	`volumes`	VOLUME_INSPECT
Block Volume	`volume-backups`	VOLUME_BACKUP_INSPECT
Compute	`console-histories`	CONSOLE_HISTORY_INSPECT
Compute	`instance-images`	INSTANCE_IMAGE_READ

Service	Resource Type	Permissions Required to View in Search Results
Compute	`instances`	INSTANCE_READ
Database	`databases`	DATABASE_INSPECT
Database	`db-homes`	DB_HOME_INSPECT (if you want to filter results using `db-homes` attributes)
Database	`db-systems`	DB_SYSTEM_INSPECT
IAM	`compartments`	COMPARTMENT_INSPECT
IAM	`groups`	GROUP_INSPECT
IAM	`identity-providers`	IDENTITY_PROVIDER_INSPECT
IAM	`users`	USER_INSPECT
Networking	`route-tables`	ROUTE_TABLE_READ
Networking	`security-lists`	SECURITY_LIST_READ
Networking	`subnets`	SUBNET_READ
Networking	`vcns`	VCN_READ
Object Storage	`buckets`	BUCKET_INSPECT

Details for the WAF Service

This topic covers details for writing policies to control access to the WAAS service.

Aggregate Resource-Type

`waas-family`

Individual Resource-Types

`waas-policy`

`waas-certificate`

`waas-work-request`

`waas-metering`

COMMENTS

A policy that uses `<verb> waas` is equivalent to writing one with a separate `<verb> <individual resource-type>` statement for each of the individual resource-types.

See the table in Details for Verb + Resource-Type Combinations for a detailed breakout of the API operations covered by each verb, for each individual resource-type included in `waas`.

Supported Variables

The WAF Service supports all the general variables (see General Variables for All Requests), plus the ones listed here.

Variable	Variable Type	Comments
`target.waas-policy.id`	Entity (OCID)	Use this variable to control access to specific WAAS policies by OCID.
`target.waf-rule-key`	String	Use this variable to control access to specific WAF rules by name.

Details for Verb + Resource-Type Combinations

The following tables show the permissions and API operations covered by each verb. The level of access is cumulative as you go from `inspect` > `read` > `use` > `manage`. A plus sign (+) in a table cell indicates incremental access compared to the cell directly above it, whereas "no extra" indicates no incremental access.

For example, the `use` and `manage` verbs for the `waas-policy` resource-type cover no extra permissions or API operations compared to the `read` verb.

waas-policy

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
WAAS_POLICY_INSPECT	`ListWaasPolicies`	*none*
	`ListWaasOriginRequestCidrs`	

READ

Permissions	APIs Fully Covered	APIs Partially Covered
WAAS_POLICY_INSPECT	`GetWaasPolicy`	*none*
WAAS_POLICY_READ		

USE

Permissions	APIs Fully Covered	APIs Partially Covered
WAAS_POLICY_INSPECT	`UpdateWaasPolicy`	*none*
WAAS_POLICY_READ		
WAAS_POLICY_UPDATE		

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
WAAS_POLICY_INSPECT	`CreateWaasPolicy`	*none*
WAAS_POLICY_READ	`DeleteWaasPolicy`	
WAAS_POLICY_UPDATE		
WAAS_POLICY_CREATE		
WAAS_POLICY_DELETE		

waas-certificate

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
WAAS_CERTIFICATE_INSPECT	`ListCertificates`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
WAAS_CERTIFICATE_INSPECT WAAS_CERTIFICATE_READ	`GetCertificate`	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
WAAS_CERTIFICATE_INSPECT WAAS_CERTIFICATE_READ	`CreateCertificate`	*none*

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
WAAS_CERTIFICATE_INSPECT WAAS_CERTIFICATE_READ WAAS_CERTIFICATE_CREATE WAAS_CERTFICATE_DELETE	`DeleteCertificate`	*none*

waas-work-request

INSPECT

Permissions	APIs Fully Covered	APIs Partially Covered
WAAS_WORK_REQUEST_INSPECT	`ListWorkRequests`	*none*

READ

Permissions	APIs Fully Covered	APIs Partially Covered
WAAS_WORK_REQUEST_INSPECT WAAS_WORK_REQUEST_READ	`GetWorkRequestDetails`	*none*

USE

Permissions	APIs Fully Covered	APIs Partially Covered
WAAS_WORK_REQUEST_INSPECT	*no extra*	*none*
WAAS_WORK_REQUEST_READ		

MANAGE

Permissions	APIs Fully Covered	APIs Partially Covered
WAAS_WORK_REQUEST_INSPECT	`DeleteWorkRequest`	*none*
WAAS_WORK_REQUEST_READ		
WAAS_WORK_REQUEST_DELETE		

waas-metering

READ

Permissions	APIs Fully Covered	APIs Partially Covered
WAAS_METERING_READ	`GetWafReport`	*none*

Permissions Required for Each API Operation

The following table lists the API operations in a logical order, grouped by resource type.

For information about permissions, see Permissions.

API Operation	Permissions Required to Use the Operation
`CreateWaasPolicy`	WAAS_POLICY_CREATE
`ListWaasPolcies`	WAAS_POLICY_INSPECT
`GetWaasPolicy`	WAAS_POLICY_READ
`UpdateWaasPolicy`	WAAS_POLICY_UPDATE
`DeleteWaasPolicy`	WAAS_POLICY_DELETE

API Operation	Permissions Required to Use the Operation
ListReports	WAAS_POLICY_INSPECT
ListWafReports	WAAS_POLICY_INSPECT
GetWafTraffic	WAAS_POLICY_READ
GetWafBlocked	WAAS_POLICY_READ
GetWafRequests	WAAS_POLICY_READ
GetWafSettings	WAAS_POLICY_READ
UpdateWafSettings	WAAS_POLICY_UPDATE
GetAccessRules	WAAS_POLICY_READ
UpdateAccessRules	WAAS_POLICY_UPDATE
GetCaptchas	WAAS_POLICY_READ
GetHumanInteractionChallenge	WAAS_POLICY_READ
UpdateHumanInteractionChallenge	WAAS_POLICY_UPDATE
GetJsChallenge	WAAS_POLICY_READ
UpdateJsChallenge	WAAS_POLICY_UPDATE
UpdateIpRateLimiting	WAAS_POLICY_UPDATE
GetGoodBots	WAAS_POLICY_READ
UpdateGoodBots	WAAS_POLICY_UPDATE
GetWafWhitelists	WAAS_POLICY_READ
GetWafRecommendations	WAAS_POLICY_READ

API Operation	Permissions Required to Use the Operation
AcceptWafRecommendations	WAAS_POLICY_UPDATE
ListWafRules	WAAS_POLICY_INSPECT
UpdateWafRuleActions	WAAS_POLICY_UPDATE
GetWafRule	WAAS_POLICY_READ
GetThreatFeeds	WAAS_POLICY_READ
UpdateThreatFeedAction	WAAS_POLICY_UPDATE
GetAlerts	WAAS_POLICY_READ
ListWorkRequests	WAAS_WORK_REQUEST_READ
ListWaasOriginRequestCidrs	WAAS_POLICY_INSPECT
GetWorkRequestDetails	WAAS_WORK_REQUEST_READ
DeleteWorkRequest	WAAS_WORK_REQUEST_DELETE
CreateCertificate	WAAS_CERTIFICATE_CREATE
ListCertificates	WAAS_CERTIFICATE_INSPECT
GetCertificate	WAAS_CERTIFICATE_READ
DeleteCertificate	WAAS_CERTIFICATE_DELETE
GetWafReport	WAAS_METERING_READ

User Credentials

There are several types of credentials that you manage with Oracle Cloud Infrastructure Identity and Access Management (IAM):

- **Console password:** For signing in to the Console, the user interface for interacting with Oracle Cloud Infrastructure. Note that federated users can't have Console passwords because they sign in through their identity provider. See Federating with Identity Providers.
- **API signing key (in PEM format):** For sending API requests, which require authentication.
- **Auth token:** An Oracle-generated token that you can use to authenticate with third-party APIs. For example, use an auth token to authenticate with a Swift client when using Recovery Manager (RMAN) to back up an Oracle Database System (DB System) database to Object Storage.
- **Customer Secret Keys:** For using the Amazon S3 Compatibility API with Object Storage. See Amazon S3 Compatibility API .
- **SMTP Credentials:** For using the Email Delivery service.



Important

API signing keys are different from the SSH keys you use to access a compute instance (see Security Credentials). For more information about API signing keys, see Required Keys and OCIDs. For more information about instance SSH keys, see Managing Key Pairs.

User Password

The administrator who creates a new user in IAM also needs to generate a one-time Console password for the user (see To create or reset another user's Console password). The administrator needs to securely deliver the password to the user by providing it verbally, printing it out, or sending it through a secure email service.

When the user signs in to the Console the first time, they'll be immediately prompted to change the password. If the user waits more than 7 days to initially sign in and change the password, it will expire and an administrator will need to create a new one-time password for the user.

Once the user successfully signs in to the Console, they can use Oracle Cloud Infrastructure resources according to permissions they've been granted through policies.



Note

A user automatically has the ability to change their password in the Console. An administrator does not need to create a policy to give a user that ability.

Changing a Password

If a user wants to change their own password *sometime after* they change their initial one-time password, they can do it in the Console. Remember that a user can automatically change *their own* password; an administrator does not need to create a policy to give the user that ability.

For more information, see To change your Console password.

If a User Needs Their Console Password Reset

If a user forgets their Console password and also has no access to the API, they can use the Console's **Forgot Password** link to have a temporary password sent to them. This option is

available if the user has an email address in their user profile.

If the user does not have an email address in their user profile, then they need to ask an administrator to reset their password for them. All administrators (and anyone else who has permission to the tenancy) can reset Console passwords. The process of resetting the password generates a new one-time password that the administrator needs to deliver to the user. The user will need to change their password the next time they sign in to the Console.

If you're an administrator who needs to reset a user's Console password, see To create or reset another user's Console password.

If a User Is Blocked from Signing In to the Console

If a user tries 10 times in a row to sign in to the Console unsuccessfully, they will be automatically blocked from further attempts. They'll need to contact an administrator to get unblocked (see To unblock a user).

API Signing Keys

A user who needs to make API requests must upload an **RSA public key in PEM format (minimum 2048 bits)** to IAM and sign the API requests with the corresponding private key (see Required Keys and OCIDs).



Important

A user automatically has the ability to upload and manage *their own* API keys in the Console or API. An administrator does not need to write a policy to give the user that ability. Remember that a user can't use the API to change or delete their own credentials until they themselves upload a key in the Console, or an administrator uploads a key for that user in the Console or the API.

If you have a non-human system that needs to make API requests, an administrator needs to create a user for that system and then upload a public key to the IAM service for the system. There's no need to generate a Console password for the user.

For instructions on uploading an API key, see To upload an API signing key.

Auth Tokens

Auth tokens are authentication tokens generated by Oracle. You use auth tokens to authenticate with third-party APIs that do not support the Oracle Cloud Infrastructure signature-based authentication, for example, the Swift API. If your service requires an auth token, the service-specific documentation instructs you to generate one and how to use it.

Federating with Identity Providers

This topic describes identity federation concepts. Oracle Cloud Infrastructure supports federation with Oracle Identity Cloud Service,and Microsoft Active Directory (via Active Directory Federation Services (AD FS)), Okta, and any identity provider that supports the Security Assertion Markup Language (SAML) 2.0 protocol.

Overview

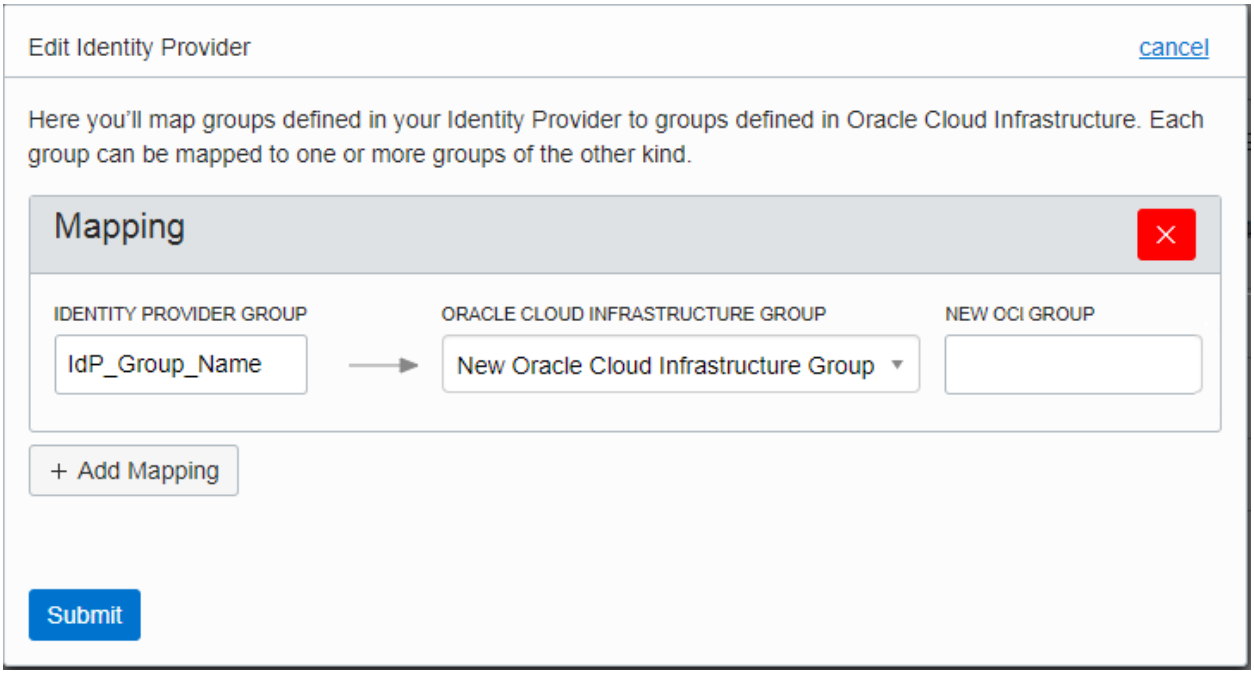
Enterprise companies commonly use an *identity provider (IdP)* to manage user login/passwords and to authenticate users for access to secure websites, services, and resources.

When someone in your company wants to use Oracle Cloud Infrastructure resources in the Console, they must sign in with a user login and password. Your administrators can federate with a supported IdP so that each employee can use an existing login and password and not have to create a new set to use Oracle Cloud Infrastructure resources.

To federate, an administrator goes through a short process to set up a relationship between the IdP and Oracle Cloud Infrastructure (commonly referred to as a *federation trust*). After an administrator sets up that relationship, any person in your company who goes to the Oracle Cloud Infrastructure Console is prompted with a "single sign-on" experience provided by the

IdP. The user signs in with the login/password that they've already set up with the IdP. The IdP authenticates the user, and then that user can access Oracle Cloud Infrastructure.

When working with your IdP, your administrator defines groups and assigns each user to one or more groups according to the type of access the user needs. Oracle Cloud Infrastructure also uses the concept of groups (in conjunction with IAM policies) to define the type of access a user has. As part of setting up the relationship with the IdP, your administrator can map each IdP group to a similarly defined IAM group, so that your company can re-use the IdP group definitions when authorizing user access to Oracle Cloud Infrastructure resources. Here's a screenshot from the mapping process:



For information about the number of federations and group mappings you can have, see Service Limits. There's no limit on the number of federated users.



Note

Any users who are in more than 50 IdP groups cannot be authenticated to use the Oracle Cloud Infrastructure Console.

Automated User Provisioning and Synchronization with SCIM

Tenancies federated with Oracle Identity Cloud Service or the third-party provider Okta, can also leverage SCIM (System for Cross-domain Identity Management) to enable provisioning of federated users in Oracle Cloud Infrastructure. Federated users that have been provisioned in Oracle Cloud Infrastructure through this process can have the additional user credentials such as API keys and auth tokens that are managed in the **User Settings** page. This enables federated users to use the SDK and CLI, and other features that require the additional user credentials. For more information, see User Provisioning for Federated Users.

General Concepts

Here's a list of the basic concepts you need to be familiar with.

IdP

IdP is short for *identity provider*, which is a service that provides identifying credentials and authentication for users.

Tenancies created after December 18, 2017 are automatically federated with Oracle Identity Cloud Service as the IdP. Oracle Cloud Infrastructure can be federated with any IdP that supports the Security Assertion Markup Language (SAML) 2.0 protocol.

SERVICE PROVIDER (SP)

A service (such as an application, website, and so on) that calls upon an IdP to authenticate users. In this case, Oracle Cloud Infrastructure is the SP.

FEDERATION TRUST

A relationship that an administrator configures between an IdP and SP. You can use the Oracle Cloud Infrastructure Console or API to set up that relationship. Then, the specific IdP is "federated" to that SP. In the Console and API, the process of federating is thought of as *adding an identity provider to the tenancy.*

SAML METADATA DOCUMENT

An IdP-provided XML-based document that provides the required information to an SP to federate with that IdP. Oracle Cloud Infrastructure supports the SAML 2.0 protocol, which is an XML-based standard for sharing required information between the IdP and SP. Depending on which idP you are federating with, you must either provide the metadata URL (see below) to this document or upload the document to Oracle Cloud Infrastructure.

METADATA URL

An IdP-provided URL that enables an SP to get required information to federate with that IdP. Oracle Cloud Infrastructure supports the SAML 2.0 protocol, which is an XML-based standard for sharing required information between the IdP and SP. The metadata URL points to the SAML metadata document the SP needs.

FEDERATED USER

Someone who signs in to use the Oracle Cloud Infrastructure Console by way of a federated IdP.

LOCAL USER

A non-federated user. In other words, someone who signs in to use the Oracle Cloud Infrastructure Console with a login and password created in Oracle Cloud Infrastructure.

GROUP MAPPING

A mapping between an IdP group and an Oracle Cloud Infrastructure group, used for the purposes of user authorization.

SCIM

SCIM (System for Cross-domain Identity Management) is an IETF standard protocol that enables user provisioning across identity systems. Oracle Cloud Infrastructure hosts a SCIM endpoint for provisioning federated users into Oracle Cloud Infrastructure. Using a SCIM client to provision users in Oracle Cloud Infrastructure enables you to assign credentials to the users in Oracle Cloud Infrastructure.

PROVISIONED (OR SYNCHRONIZED) USER

A user provisioned by the identity provider's SCIM client in Oracle Cloud Infrastructure. These users can be listed in the Oracle Cloud Infrastructure Console and can have all the Oracle Cloud Infrastructure user credentials except for a Console password.

ENCRYPT ASSERTION

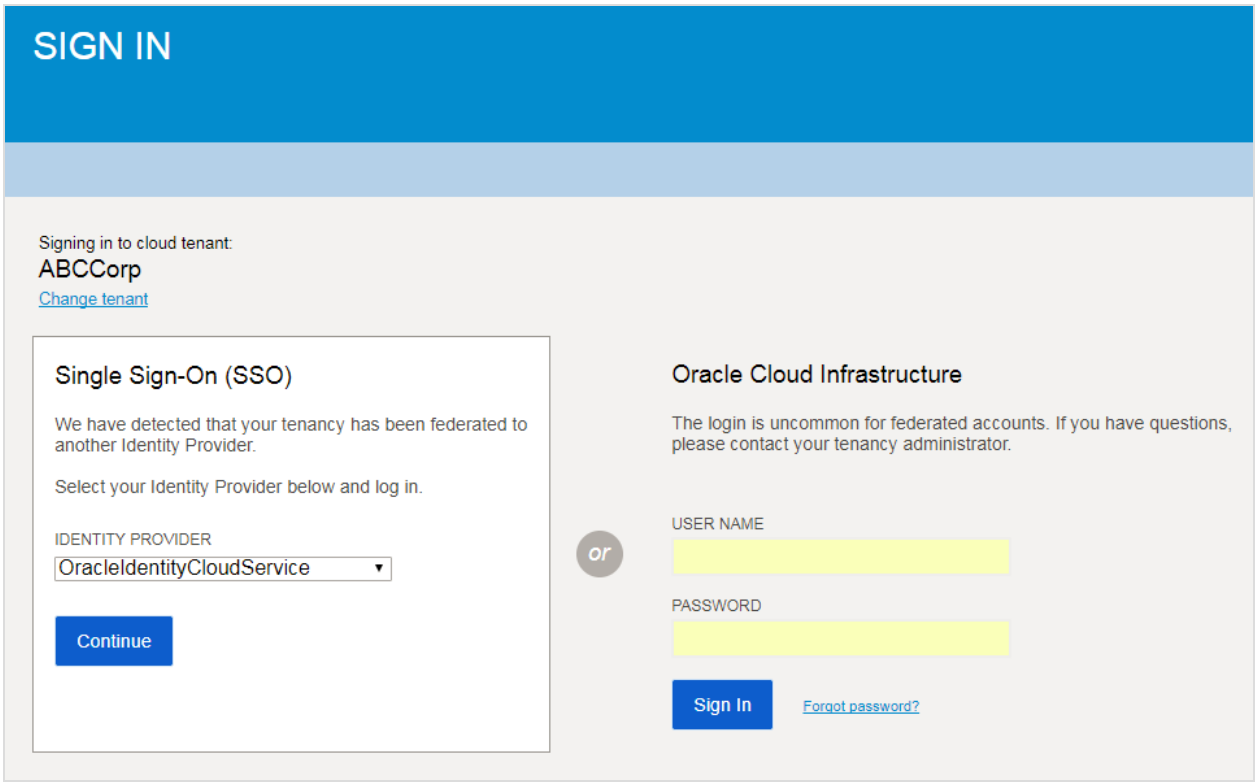
Some IdPs support the encryption of the SAML assertion. When enabled, the service provider expects the SAML assertion to be encrypted by the identity provider, using the service provider's encryption key. In this case, the service provider is Oracle Cloud Infrastructure authentication service. If you choose to enable this feature of your IdP, you must also enable the feature when you set up your Federation provider in the IAM service. Note that Microsoft AD FS enables the encryption of the SAML assertion by default. If your IdP is Microsoft AD FS, you must either enable this feature in IAM or disable it for Microsoft AD FS.

Experience for Federated Users

Federated users can use the Console to access Oracle Cloud Infrastructure (according to IAM policies for the groups the users are in).

They'll be prompted to enter their Oracle Cloud Infrastructure tenant (for example, ABCCorp).

They then see a page with two sets of sign-in instructions: one for federated users and one for non-federated (Oracle Cloud Infrastructure) users. See the following screenshot.



The tenant name is shown on the left. Directly below is the sign-in area for federated users. On the right is the sign-in area for non-federated users.

Federated users choose which identity provider to use for sign-in, and then they're redirected to that identity provider's sign-in experience for authentication. After entering their login and password, they are authenticated by the IdP and redirected back to the Oracle Cloud Infrastructure Console.

The federated users (without SCIM configuration) cannot access the "User Settings" page in the Console. This page is where a user can change or reset their Console password and manage other Oracle Cloud Infrastructure credentials such as API signing keys and auth tokens.

Experience for Federated Users with SCIM Configuration

If your IdP has also been configured with a SCIM client, a user signed in through their identity provider can access the User Settings page and have user capabilities such as API keys, auth tokens, and other user credentials. (**Note:** This is currently available for Oracle Identity Cloud Service and Okta federations only.)

Required IAM Policy

To add and manage identity providers in your tenancy, you must be authorized by an IAM policy. If you're in the Administrators group, then you have the required access.

Here's a more limited policy that restricts access to only the resources related to identity providers and group mappings:

```
Allow group IdPAdmins to manage identity-providers in tenancy

Allow group IdPAdmins to manage groups in tenancy
```

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for groups or other IAM components, see Details for IAM.

Supported Identity Providers



Important

Oracle Cloud Infrastructure tenancies created December 18, 2017 or later are automatically federated with Oracle Identity Cloud Service.

If your tenancy was created before December 18, 2017, and you want to set up a federation with Oracle Identity Cloud Service, see Federating with Oracle Identity Cloud Service.

For instructions for federating with other identity providers, see the following:

Federating with Microsoft Active Directory

Cloud Infrastructure Okta Configuration for Federation and Provisioning (white paper)

Federating with SAML 2.0 Identity Providers

Federating with Oracle Identity Cloud Service

This topic points to the appropriate topics for federating Oracle Cloud Infrastructure with Oracle Identity Cloud Service depending on when you activated your tenancy.

Tenancies created December 21, 2018 and after

These tenancies are automatically federated with Oracle Identity Cloud Service and configured to provision federated users in Oracle Cloud Infrastructure.

To manage your federated users and groups, see Managing Oracle Identity Cloud Service Users and Groups in the Oracle Cloud Infrastructure Console.

For information about the federation, see Frequently Asked Questions for Oracle Identity Cloud Service Federated Users.

Tenancies created between December 18, 2017 and December 20, 2018

These tenancies are automatically federated with Oracle Identity Cloud Service but are not configured to provision federated users in Oracle Cloud Infrastructure to allow these users to have additional credentials (API keys, auth tokens, etc.).

To enable this feature for users, you need to perform a one-time upgrade, see: User Provisioning for Federated Users.

After you have performed this upgrade, see Managing Oracle Identity Cloud Service Users and Groups in the Oracle Cloud Infrastructure Console to manage your federated users and groups.

Tenancies created before December 18, 2017

These tenancies must be manually federated with Oracle Identity Cloud Service. See Federating with Oracle Identity Cloud Service described below.

Manually Federating with Oracle Identity Cloud Service

Your organization can have multiple Oracle Identity Cloud Service accounts (e.g., one for each division of the organization). You can federate multiple Identity Cloud Service accounts with Oracle Cloud Infrastructure, but each federation trust that you set up must be for a single Identity Cloud Service account.



Note

Before following the steps in this topic, see Federating with Identity Providers to ensure that you understand general federation concepts.

Web Application and Client Credentials

For each trust, you must set up a *web application* in Oracle Identity Cloud Service (also called a *trusted application*); instructions are in Instructions for Federating with Oracle Identity Cloud Service. The resulting application has a set of client credentials (a client ID and client secret). When you federate your Identity Cloud Service account with Oracle Cloud Infrastructure, you must provide these credentials.

COMPUTEBAREMETAL application

A *trusted application* in Oracle Identity Cloud Service that contains the set of client credentials (a client ID and client secret) you'll need to provide when you federate your Identity Cloud Service account with Oracle Cloud Infrastructure.

Required URLs

The easiest way to federate with Oracle Identity Cloud Service is through the Oracle Cloud Infrastructure Console, although you could do it programmatically with the API. If you're

using the Console, you're asked to provide a *base URL* instead of the metadata URL. The base URL is the left-most part of the URL in the browser window when you're signed in to the Identity Cloud Service console:

- **Base URL:** *<Identity Cloud Service account name>*.identity.oraclecloud.com

If you're using the API to federate, you need to provide the metadata URL, which is the base URL with /fed/v1/metadata appended, like so:

- **Metadata URL:** *<Identity Cloud Service account name>*.identity.oraclecloud.com/fed/v1/metadata

The metadata URL links directly to the IdP-provided XML required to federate. If you're using the API, you need to provide both the metadata URL and the metadata itself when federating. For more information, see Managing Identity Providers in the API.

OCI-V2-<TENANCY_NAME> APP

When you manually federate an Oracle Identity Cloud Service account with Oracle Cloud Infrastructure, a new SAML application called *OCI-V2-<tenancy_name>* is automatically created in that Oracle Identity Cloud Service account. If you later need to delete the Oracle Identity Cloud Service identity provider from your Oracle Cloud Infrastructure tenancy, make sure to also delete the OCI-V2-*<tenancy_name>* from Oracle Identity Cloud Service. If you don't, and you later try to federate the same Oracle Identity Cloud Service account again, you'll get a 409 error saying that an application with the same name already exists (that is, OCI-V2-*<tenancy_name>*).

PROVISIONED USER

A provisioned user is provisioned by Oracle Identity Cloud Service in Oracle Cloud Infrastructure and is synched to a federated user that is managed in Oracle Identity Cloud Service. The provisioned user can have the special Oracle Cloud Infrastructure credentials like API keys and auth tokens to enable programmatic access. Provisioned users cannot have Console passwords.

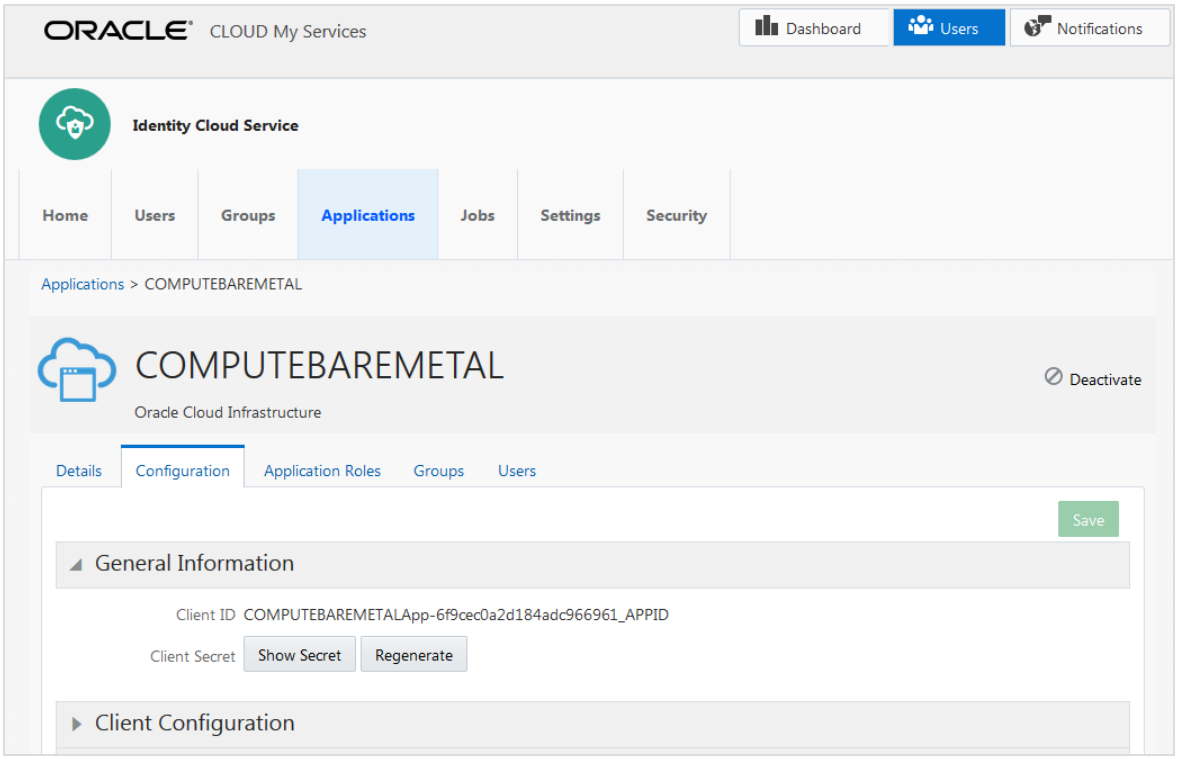
Instructions for Federating with Oracle Identity Cloud Service

Following is the general process an administrator goes through to set up the identity provider, and below are instructions for each step. It's assumed that the administrator is an Oracle Cloud Infrastructure user with the required credentials and access.

1. In Oracle Identity Cloud Service, get the required information you'll need to perform the set up steps in Oracle Cloud Infrastructure.
2. In Oracle Cloud Infrastructure, set up the federation:
    a. Set up Oracle Identity Cloud Service as an identity provider.
    b. Map Oracle Identity Cloud Service groups to IAM groups.
3. In Oracle Cloud Infrastructure, set up the IAM policies for the IAM groups to define the access you want the members of the mapped groups to have.
4. Inform your users of the name of your Oracle Cloud Infrastructure tenant and the URL for the Console (for example, https://console.us-ashburn-1.oraclecloud.com).

Step 1: Get required information from Oracle Identity Cloud Service

1. Go to the Oracle Identity Cloud Service console and sign in with admin privileges. Make sure you're viewing the Admin Console.
2. In the Identity Cloud Service console, click **Applications**. The list of trusted applications is displayed.
3. Click COMPUTEBAREMETAL.
4. Click **Configuration**.
5. Expand **General Information**. The client ID is displayed. Click **Show Secret** to display the client secret.



6. Record the Client ID and Client Secret. They look similar to this:
    - Client ID: de06b81cb45a45a8acdcde923402a9389d8
    - Client Secret: 8a297afd-66df-49ee-c67d-39fcdf3d1c31

Step 2: Add Oracle Identity Cloud Service as an identity provider in Oracle Cloud Infrastructure

1. Go to the Console and sign in with your Oracle Cloud Infrastructure login and password.
2. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
3. Click **Add identity provider**.

4. Enter the following:
    a. **Name:** A unique name for this federation trust. This is the name federated users see when choosing which identity provider to use when signing in to the Console (for example., *ABCCorp_IDCS* as shown in the screenshot in Experience for Federated Users). The name must be unique across all identity providers you add to the tenancy. You cannot change this later.
    b. **Description:** A friendly description.
    c. **IDCS Base URL:** See Required URLs.
    d. **Client ID:** From Step 1: Get required information from Oracle Identity Cloud Service.
    e. **Client secret:** From Step 1: Get required information from Oracle Identity Cloud Service.
    f. Click **Show Advanced Options**.
    g. **Encrypt Assertion:** Selecting the check box lets the IAM service know to expect the encryption from the IdP. If you select this check box, you must also set up encryption of the assertion in IDCS. For more information, see Encrypt Assertion. For information about setting this feature up in the IDCS, see Managing Oracle Identity Cloud Service Applications.
    h. Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
5. Click **Continue**.
6. Set up the mappings between Oracle Identity Cloud Service groups and IAM groups in Oracle Cloud Infrastructure. A given Oracle Identity Cloud Service group can be mapped to zero, one, or multiple IAM groups, and vice versa. However, each individual mapping is between only a single Oracle Identity Cloud Service group and a single IAM group. Changes to group mappings take effect typically within seconds.



Note

If you don't want to set up the group mappings now, you can simply click **Create** and come back to add the mappings later.

To create a group mapping:

a. Select the Oracle Identity Cloud Service group from the list under **Identity Provider Group**.

b. Select the IAM group you want to map from the list under **Oracle Cloud Infrastructure Group**. If you instead want to create a new IAM group, select **New OCI Group** and enter the name of the new group in **New OCI Group Name**. The new group is automatically created in IAM and mapped to the IdP group. It will also automatically be given this description, which you can't change: "Group created during federation".



Tip

Requirements for IAM group name: No spaces. Allowed characters: letters, numerals, hyphens, periods, underscores, and plus signs (+). The name cannot be changed later.

c. Repeat the above sub-steps for each mapping you want to create, and then click **Create**.

AFTER THE FEDERATION SET UP

The identity provider is now added to your tenancy and appears in the list on the **Federation** page. Click the identity provider to view its details and the group mappings you just set up.

Oracle assigns the identity provider and each group mapping a unique ID called an Oracle Cloud ID (OCID). For more information, see Resource Identifiers.

In the future, come to the **Federation** page if you want to edit the group mappings or delete the identity provider from your tenancy.

Users that are members of the Oracle Identity Cloud Service groups mapped to the Oracle Cloud Infrastructure groups are now listed in the Console on the Users page. See Managing User Capabilities for Federated Users for more information on assigning these users additional credentials.

Step 3: Set up IAM policies for the groups

If you haven't already, set up IAM policies to control the access the federated users have to your organization's Oracle Cloud Infrastructure resources. For more information, see Getting Started with Policies and Common Policies.

Step 4: Give your federated users the name of the tenant and URL to sign in

The federated users need the URL for the Oracle Cloud Infrastructure Console (for example, https://console.us-ashburn-1.oraclecloud.com) and the name of your tenant. They'll be prompted to provide the tenant name when they sign in to the Console.

Managing Identity Providers in the Console

To add an Oracle Identity Cloud Service as an identity provider

See Instructions for Federating with Oracle Identity Cloud Service.

To delete the identity provider

All the group mappings will also be deleted.

1. Delete the identity provider from your tenancy:
    a. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
    A list of the identity providers in your tenancy is displayed.
    b. Click the identity provider to view its details.
    c. Click **Delete**.
    d. Confirm when prompted.
2. Delete the *OCI-V2-<tenancy_name>* from your Oracle Identity Cloud Service account:
    a. Go to Oracle Identity Cloud Service and sign in to the federated account.
    b. Click **Applications**. The list of applications is displayed.
    c. Locate the *OCI-V2-<tenancy_name>* and click its name to view its details page.
    d. In the upper right of the page, click **Deactivate**. Confirm when prompted.
    e. Click **Remove**. Confirm when prompted.

To add group mappings for Oracle Identity Cloud Service

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
    A list of the identity providers in your tenancy is displayed.
2. Click the name you chose for your Oracle Identity Cloud Service federation to view its details.
3. Click **Edit Provider Details**.
4. Add at least one mapping:
    a. Click **+ Add Mapping**.
    b. Select the Oracle Identity Cloud Service group from the list under **Identity Provider Group**.
    c. Select the IAM group you want to map from the list under **Oracle Cloud**

Infrastructure Group. If you instead want to create a new IAM group, select **New OCI Group** and enter the name of the new group in **New OCI Group Name**. The new group is automatically created in IAM and mapped to the IdP group. It will also automatically be given this description, which you can't change: "Group created during federation".

d. Repeat the above sub-steps for each mapping you want to create, and then click **Submit**.

Your changes take effect typically within seconds in your home region. Wait several more minutes for changes to propagate to all regions.

Users that are members of the Oracle Identity Cloud Service groups mapped to the Oracle Cloud Infrastructure groups are now listed in the Console on the Users page. See Managing User Capabilities for Federated Users for more information on assigning these users additional credentials.

To update or delete a group mapping

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
   A list of the identity providers in your tenancy is displayed.
2. Click the identity provider to view its details.
3. Click **Edit Mapping**.
4. Update the mappings (or click the X to delete a mapping), and then click **Submit**.

Your changes take effect typically within seconds in your home region. Wait several more minutes for changes to propagate to all regions.

If this action results in federated users no longer having membership in any group that is mapped to Oracle Cloud Infrastructure, the federated users' provisioned users' will also be removed from Oracle Cloud Infrastructure. Typically, this process takes several minutes.

Managing Identity Providers in the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations:

Identity providers:

- CreateIdentityProvider
- ListIdentityProviders
- GetIdentityProvider
- UpdateIdentityProvider
- DeleteIdentityProvider: Before you can use this operation, you must first use DeleteIdpGroupMapping to remove all the group mappings for the identity provider.

Group mappings:

- CreateIdpGroupMapping: Each group mapping is a separate entity with its own OCID.
- ListIdpGroupMappings
- GetIdpGroupMapping
- UpdateIdpGroupMapping
- DeleteIdpGroupMapping

Managing Oracle Identity Cloud Service Users and Groups in the Oracle Cloud Infrastructure Console

This topic describes how to use the Oracle Cloud Infrastructure Console to manage your Oracle Identity Cloud Service users and groups. Before you get started, understand basic federation concepts. See Federating with Identity Providers.

Working with Oracle Identity Cloud Service in the Console

The Oracle Cloud Infrastructure Console provides an integration with Oracle Identity Cloud Service (IDCS) that lets you perform some basic management of your IDCS users and groups

right within the Console.

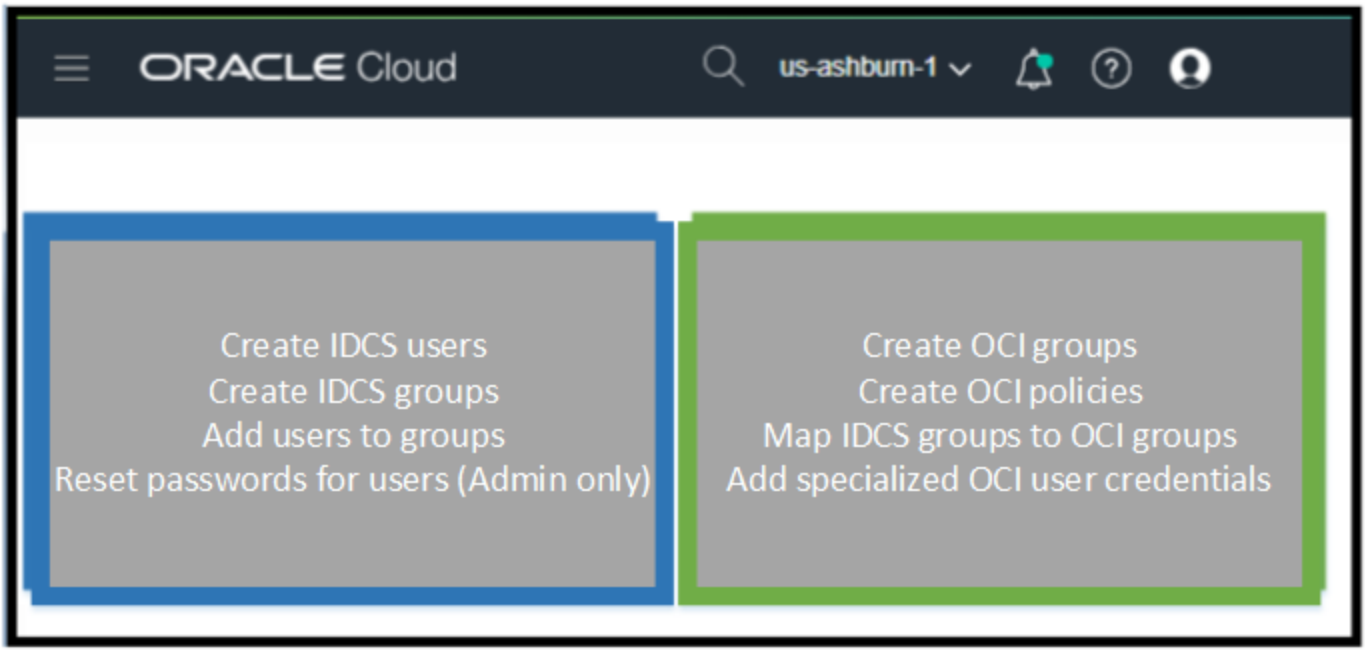
In the Console, you can create your Oracle Identity Cloud Service groups and users that you want to grant access to Oracle Cloud Infrastructure resources.

Remember that for the members of a group in Oracle Identity Cloud Service to have permissions in Oracle Cloud Infrastructure, you must map the group to a group in Oracle Cloud Infrastructure. Before you set up any new groups in Oracle Identity Cloud Service, ensure that you understand how to assign permissions to groups in Oracle Cloud Infrastructure. See Overview of Oracle Cloud Infrastructure Identity and Access Management.

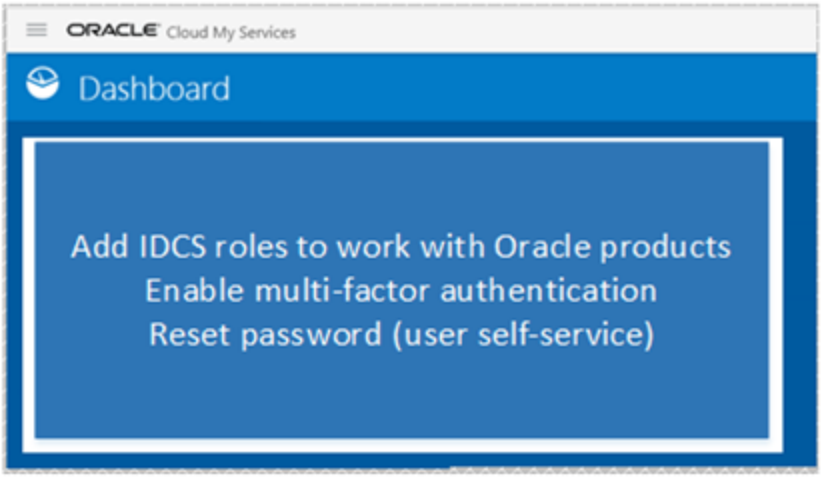
Although the Console supports many basic user and group management tasks, some tasks still require you to switch to the My Services dashboard.

The following graphic summarizes the tasks you can perform in each interface:

Perform the following tasks in the Oracle Cloud Infrastructure Console:



Perform these tasks in My Services Console:



Required Policies and Permissions

To manage Oracle Identity Cloud Service users and groups in the Oracle Cloud Console, you'll need to be granted permissions in both the Oracle Cloud Infrastructure IAM service and in Oracle Identity Cloud Service.

Members of the OCI_Administrators group have the required permissions to create groups and policies in Oracle Cloud Infrastructure.

Important: To create users and groups in the Oracle Identity Cloud Service federation, you'll need the Identity Domain Administrator role, or be a member of a group that has been granted that role. For information on Oracle Identity Cloud Service roles, see Administering Oracle Identity Cloud Service.

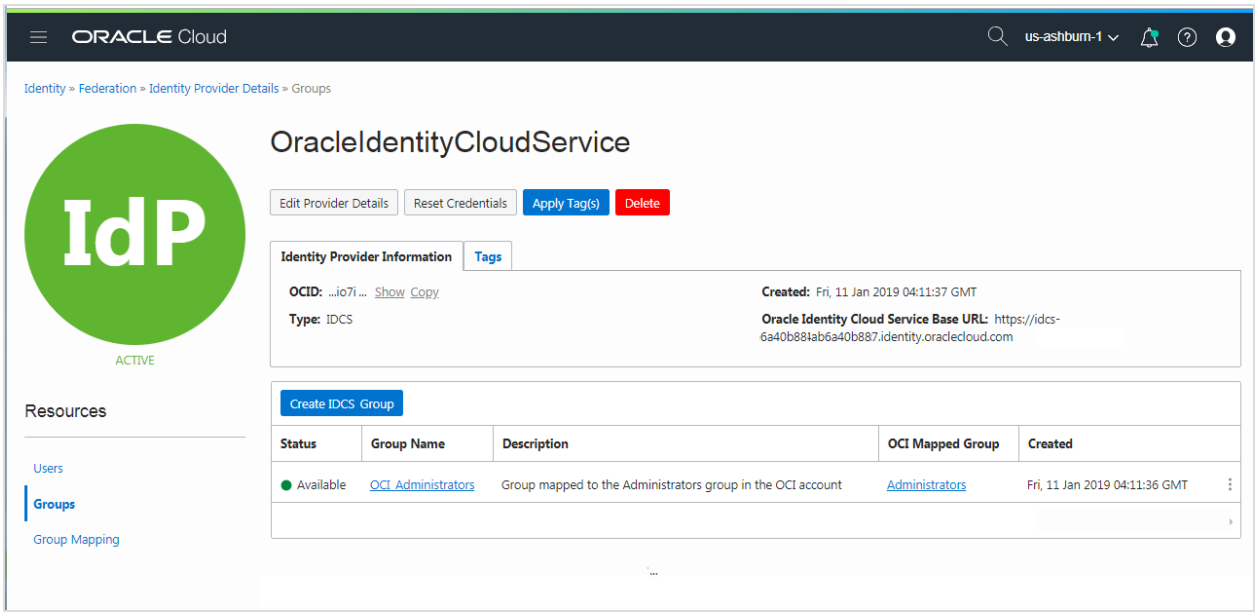
To quickly create a user with the required permissions, see Add a User with Oracle Cloud Administrator Permissions.

Navigating to Your Oracle Identity Cloud Service Users and Groups in the Console

In the Console, you can add users and groups to Oracle Identity Cloud Service from the Identity Provider Details page.

To view your identity provider details:

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
2. Click your Oracle Identity Cloud Service federation. For most tenancies, the federation is named **OracleIdentityCloudService**.
   The identity provider details page is displayed.



From the Identity Provider Details page, click **Users** to display the users that exist in Oracle Identity Cloud Service. Click **Groups** to display the groups that exist in Oracle Identity Cloud Service.

WORKING WITH ORACLE IDENTITY CLOUD SERVICE GROUPS

The Console displays all your Oracle Identity Cloud Service groups whether they are mapped to Oracle Cloud Infrastructure groups or not.

The Oracle Cloud Infrastructure Console lets you perform the following tasks to manage groups in Oracle Identity Cloud Service:

- Add groups
- Delete groups
- Edit the name and description
- Add users to groups
- Remove users from groups
- Map groups to Oracle Cloud Infrastructure groups

Some tasks you can't perform in the Oracle Cloud Infrastructure Console. To enable access to other Oracle Cloud products, you still need to assign roles in the My Services console. For more information about using Oracle Identity Cloud Service, see Administering Oracle Identity Cloud Service.

WORKING WITH ORACLE IDENTITY CLOUD SERVICE USERS

The Oracle Cloud Infrastructure Console lets you perform the following tasks to manage users in Oracle Identity Cloud Service:

- Add users
- Delete users
- Edit user details
- Add users to groups
- Remove users from groups
- Administrator reset password for other users

Notice that you can view all your Oracle Identity Cloud Service users in the Console, even users that do not have permissions in Oracle Cloud Infrastructure.

USER MANAGEMENT TASKS YOU CAN'T PERFORM IN THE CONSOLE

The Oracle Cloud Console does not support management of the following Oracle Identity Cloud Service user features and tasks:

- Manage multi-factor authentication
- Grant roles to use Oracle Cloud products
- User self-service password reset

For information about managing these tasks, see Administering Oracle Identity Cloud Service.

MANAGING ORACLE IDENTITY CLOUD SERVICE GROUPS IN THE CONSOLE

To create a group in Oracle Identity Cloud Service

This procedure creates a new group in Oracle Identity Cloud Service. Optionally, you can add users to the group at the time you create it. This group will not have any permissions in Oracle Cloud Infrastructure until you map it to an Oracle Cloud Infrastructure group.

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
   A list of the federations in your tenancy is displayed.
2. Click your Oracle Identity Cloud Service federation. For most tenancies, the federation is named **OracleIdentityCloudService**.
   The identity provider details page is displayed.
3. Under **Resources**, click **Groups**.
   The list of existing groups is displayed.
4. Click **Create IDCS Group**.
5. Enter the following:
   - **Name:** A unique name for the group.
   - **Description:** A friendly description. You can change this later if you want to.
   - **Users:** Add Oracle Identity Cloud Service users to this group. You can add users when you create the group, or later. Select users from the list. To find a specific user, you can start typing the user name to filter the list as you type.
6. Click **Create**.

After you create a group in Oracle Identity Cloud Service, you'll want to map it to an Oracle Cloud Infrastructure group as described in the next procedure.

To map an Oracle Identity Cloud Service group to an Oracle Cloud Infrastructure group

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
2. Click your Oracle Identity Cloud Service federation. For most tenancies, the federation is named **OracleIdentityCloudService**.
   The identity provider details page is displayed.
3. Click **Edit Mapping**.
4. In the **Edit Identity Provider** dialog, click **+ Add Mapping**.
5. Select the **Identity Provider Group** you want to map from the list. To find a group without scrolling through the list, you can start typing the group name to filter the list as you type.
6. Select the **OCI Group** you want to map this Identity Cloud Service group to. To find a group without scrolling through the list, you can start typing the group name to filter the list as you type.
7. To add more mappings, click **+ Add Mapping** and continue adding the mappings.
8. Select the group you want to map this group to from the list under **OCI Mapped User Group**.

Members of this group now have the permissions granted to the OCI Mapped User Group.

To edit an Oracle Identity Cloud Service group

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
2. Click your Oracle Identity Cloud Service federation. For most tenancies, the federation is named **OracleIdentityCloudService**.
   The identity provider details page is displayed.
3. Under **Resources**, click **Groups**.

The list of existing groups in the federation is displayed.

4. Find the group you want to edit and click its name.
   The **Group Details** page is displayed.
5. Click **Edit**.
6. You can update the **Group Name** or the **Description**.
7. Click **Update** to save your changes.



Warning

Changing the group name will break mappings to Oracle Cloud Infrastructure (OCI) groups. If you change the group name, ensure that you delete any existing group mappings and add new mappings with the new name. See the previous task on editing mappings.

To add existing users to a group

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
2. Click your Oracle Identity Cloud Service federation. For most tenancies, the federation is named **OracleIdentityCloudService**.
   The identity provider details page is displayed.
3. Under **Resources**, click **Groups**.
   The list of existing groups is displayed.
4. Find the group you want add a user to.
   The **User Group Details** page is displayed.
5. Click **Add IDCS User**.
6. Select the user you want to add to this group from the **Users** list.

7. Click **Add**.

To remove users from a group

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
2. Click your Oracle Identity Cloud Service federation. For most tenancies, the federation is named **OracleIdentityCloudService**.
   The identity provider details page is displayed.
3. Under **Resources**, click **Groups**.
   The list of existing groups is displayed.
4. Find the group you want to remove the user from.
   The list of users is displayed in the **Group Details** page.
5. Find the user you want to remove, and then click the the Actions icon (three dots).
6. Click **Remove User**.
7. Confirm when prompted.

To delete a group

Note: To delete a group, you must first remove all the users.

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
2. Click your Oracle Identity Cloud Service federation. For most tenancies, the federation is named **OracleIdentityCloudService**.
   The identity provider details page is displayed.
3. Under **Resources**, click **Groups**.
   The list of existing groups is displayed.
4. Find the group you want to edit and click its name.

The **Group Details** page is displayed.

5. Click **Delete**.
6. Confirm when prompted.

Create a policy to grant the group permissions on Oracle Cloud Infrastructure resources

The group you created in Oracle Identity Cloud Service gets permissions to access resources in Oracle Cloud Infrastructure through the policy you assign to the Oracle Cloud Infrastructure group. Before you complete this step, you need to decide what permissions you want to give your new group. For more information, see Getting Started with Policies and Common Policies.

Prerequisite: The group and compartment that you're writing the policy for must already exist.

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Policies**.
   A list of the policies in the compartment you're viewing is displayed.
2. If you want to attach the policy to a compartment other than the one you're viewing, select the desired compartment from the list on the left. Where the policy is attached controls who can later modify or delete it (see Policy Attachment).
3. Click **Create Policy**.
4. Enter the following:
   - **Name:** A unique name for the policy. The name must be unique across all policies in your tenancy. You cannot change this later.
   - **Description:** A friendly description. You can change this later if you want to.
   - **Policy Versioning:** Select **Keep Policy Current** if you'd like the policy to stay current with any future changes to the service's definitions of verbs and resources. Or if you'd prefer to limit access according to the definitions that were

current on a specific date, select **Use Version Date** and enter that date in format YYYY-MM-DD format. For more information, see Policy Language Version.

- **Statement:** A policy statement. For the correct format to use, see Policy Basics and also Policy Syntax. If you want to add more than one statement, click **+**.
  For example:
  To allow your group to manage all resources within a specified compartment enter a statement like the following:

  ```
  Allow group <OCI_group_name> to manage all-resources in compartment <compartment_name>
  ```

  For more policy examples, see Common Policies.
- **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

5. Click **Create**.

MANAGING ORACLE IDENTITY CLOUD SERVICE USERS IN THE CONSOLE

After you add a user in Oracle Identity Cloud Service, a user is also automatically provisioned in Oracle Cloud Infrastructure. This provisioned user can have the Oracle Cloud Infrastructure credentials, such as API keys and auth tokens.. To understand this provisioning, see User Provisioning for Federated Users.

To create a user

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
2. Click your Oracle Identity Cloud Service federation. For most tenancies, the federation is named **OracleIdentityCloudService**.

The identity provider details page is displayed.

3. Click **Create IDCS User**.
4. In the **Create IDCS User** dialog enter the following:
   - **User Name**: Enter a unique name or email address for the new user. The value will be the user's login to the Console and must be unique across all other users in your tenancy.
   - **Email**: Enter an email address for this user. The initial sign-in credentials will be sent to this email address.
   - **First Name:** Enter the user's first name.
   - **Last Name:** Enter the user's last name.
   - **Phone Number:** Optionally, enter a phone number.
   - **Groups:** Optionally, select groups to add this user to.
5. Click **Create User**.



Important

For the user to have permissions in Oracle Cloud Infrastructure, you must assign the user to a group that is mapped to an Oracle Cloud Infrastructure group. Or, if you are also creating a new group, you can perform this mapping later. The user will not be able to sign in to the Console until the mapping is accomplished.

The user creation process generates an email that is sent to the address provided that you entered. The email includes the new user's login and password to use with the Oracle Cloud InfrastructureConsole.

To add API keys, auth tokens, customer secret keys, or SMTP credentials for this user, see Managing User Capabilities for Federated Users.

To delete a user

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
2. Click your Oracle Identity Cloud Service federation. For most tenancies, the federation is named **OracleIdentityCloudService**.
   The identity provider details page is displayed.
3. Under **Resources**, click **Users**.
   The list of existing user groups in the federation is displayed.
4. Find the user you want to delete and click the name.
   The **User Details** page is displayed.
5. Click **Delete**.

To edit a user

You can update the following fields:

- Email address
- First and last name
- Phone number

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
2. Click your Oracle Identity Cloud Service federation. For most tenancies, the federation is named **OracleIdentityCloudService**.
   The identity provider details page is displayed.
3. Under **Resources**, click **Users**.
   The list of existing users is displayed.
4. Find the user you want to edit and click its name.

The **User Details** page is displayed.

5. Click **Edit**.
6. Update the fields.
7. Click **Save** when finished.

To reset a user's password

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
2. Click your Oracle Identity Cloud Service federation. For most tenancies, the federation is named **OracleIdentityCloudService**.
   The identity provider details page is displayed.
3. Under **Resources**, click **Users**.
   The list of existing user groups in the federation is displayed.
4. Find the user you want to reset the password for and click the name.
   The **User Details** page is displayed.
5. Click **Reset Password**.
   The user's password is reset. This user can't access their account until they complete the password reset steps.
6. Click **Email Password Instructions** to send the password link and instructions to the user.
   The password link is good for 24 hours. If the user does not reset their password in time, you can generate a new password link by clicking **Reset Password** for the user again.

To add API keys, auth tokens, or other Oracle Cloud Infrastructure credentials

1. View the user's details:

- If you're adding credentials for *yourself*: Open the **User** menu () and click **User Settings**.
- If you're an administrator adding credentials for *another user*: Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
  Click your Oracle Identity Cloud Service federation. For most tenancies, the federation is named **OracleIdentityCloudService**.
  The identity provider details page is displayed.
  Find the user in the list and click the **OCI Synched User** link.

2. Add the credentials for the user.

For more details about these credentials, see Managing User Credentials.

MANAGING GROUP MAPPINGS

To add group mappings for Oracle Identity Cloud Service

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
   A list of the identity providers in your tenancy is displayed.
2. Click your Oracle Identity Cloud Service federation. For most tenancies, the federation is named **OracleIdentityCloudService**.
   The identity provider details page is displayed.
3. Click **Edit Provider Details**.
4. Add at least one mapping:
   a. Click **+ Add Mapping**.
   b. Select the Oracle Identity Cloud Service group from the list under **Identity Provider Group**.
   c. Select the IAM group you want to map from the list under **Oracle Cloud**

Infrastructure Group. If you instead want to create a new IAM group, select **New OCI Group** and enter the name of the new group in **New OCI Group Name**. The new group is automatically created in IAM and mapped to the IdP group. It will also automatically be given this description, which you can't change: "Group created during federation".

d. Repeat the above sub-steps for each mapping you want to create, and then click **Submit**.

Your changes take effect typically within seconds in your home region. Wait several more minutes for changes to propagate to all regions.

Users that are members of the Oracle Identity Cloud Service groups mapped to the Oracle Cloud Infrastructure groups are now listed in the Console on the Users page. See Managing User Capabilities for Federated Users for more information on assigning these users additional credentials.

To update or delete a group mapping

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
   A list of the identity providers in your tenancy is displayed.
2. Click the identity provider to view its details.
3. Click **Edit Mapping**.
4. Update the mappings (or click the X to delete a mapping), and then click **Submit**.

If this action results in federated users no longer having membership in any group that is mapped to Oracle Cloud Infrastructure, the federated users' provisioned users' will also be removed from Oracle Cloud Infrastructure. Typically, this process takes several minutes.

Frequently Asked Questions for Oracle Identity Cloud Service Federated Users

When you sign up for Oracle Cloud Infrastructure, your account is automatically federated with Oracle Identity Cloud Service as your identity provider. This topic answers some

frequently asked questions about the federation.

What resources are created in Oracle Identity Cloud Service?

THE FOLLOWING RESOURCES ARE CREATED IN IDENTITY CLOUD SERVICE:

- **Applications:**
  - *OCI-V2-<tenancy_name>*
    This SAML application that creates the federation with Oracle Cloud Infrastructure.
  - COMPUTEBAREMETAL application
    A supporting application for the federation.

    Important

    Do not delete these applications.

- **Group:**
  OCI_Administrators group
  This group is mapped to the Administrators group in Oracle Cloud Infrastructure. Members of this group have full administrator privileges in Oracle Cloud Infrastructure.
- **User:**
  A default administrator user (e.g., user@example.com) who is a member of the OCI_ Administrators group.

What resources are created in Oracle Cloud Infrastructure?

THE FOLLOWING RESOURCES ARE CREATED IN ORACLE CLOUD INFRASTRUCTURE:

- **Identity Provider:** OracleIdentityCloudService
- **Group Mappings:**The federation is created with one group mapping:

OCI_Administrators group (from Oracle Identity Cloud Service) is mapped to the Administrators group (In Oracle Cloud Infrastructure).

- **Users:**
  - The default administrator user created in Oracle Identity Cloud Service is provisioned in Oracle Cloud Infrastructure. This user can have the Oracle Cloud Infrastructure credentials, but not a Console password.
  - A default administrator local-user with the same user name (user@example.com) is also created in Oracle Cloud Infrastructure's IAM service. Customers who choose **not** to use the Oracle Identity Cloud Service federation can use this user to administer Oracle Cloud Infrastructure.



Important

The default administrator created in Oracle Identity Cloud Service and the local default administrator created in Oracle Cloud Infrastructure exist independently in their respective identity systems. Ensure that you manage passwords for them separately.

Why is my account federated with Oracle Identity Cloud Service?

Oracle My Services links to multiple Oracle services, and so uses an Oracle-wide identity solution. Your tenant administrator account is automatically federated with Oracle Identity Cloud Service, the identity provider for many Oracle services. Federating Oracle Cloud Infrastructure with Oracle Identity Cloud Service allows you to have a seamless connection between services, without having to create a separate username and password for each one.

How do I add a user to Oracle Identity Cloud Service (a federated user)?

See Managing Oracle Identity Cloud Service Users and Groups in the Oracle Cloud

Infrastructure Console.

Can I add a user just for Oracle Cloud Infrastructure?

Yes. If you don't want to manage the user in Oracle Identity Cloud Service, you can add a user directly to the Oracle Cloud Infrastructure IAM service. See Adding Users. Using this procedure, you can create users who can sign in directly to the Oracle Cloud Infrastructure Console. Users created with this procedure do not have access to any other Oracle services.

How do I manage groups?

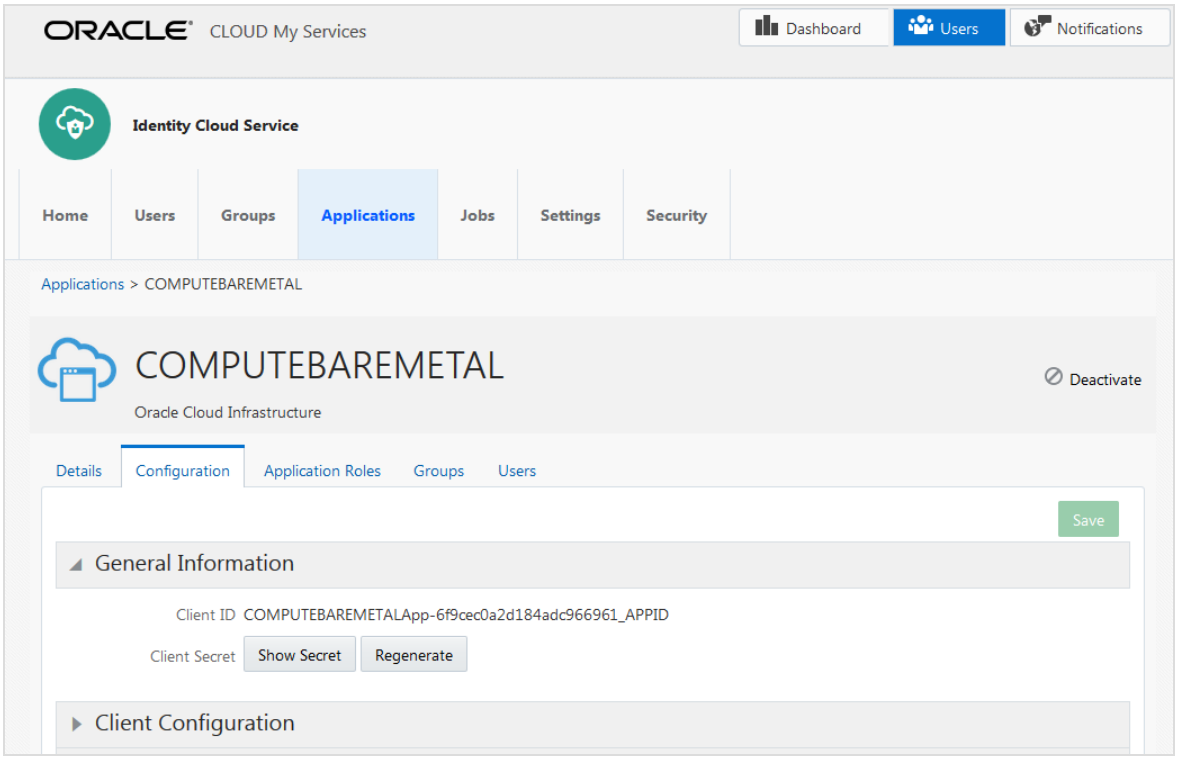
In short, managing groups requires actions in both Oracle Identity Cloud Service and Oracle Cloud Infrastructure. Groups you create in Oracle Identity Cloud Service have no privileges in Oracle Cloud Infrastructure until you map them to a group in Oracle Cloud Infrastructure. You define the policies that permit access to Oracle Cloud Infrastructure resources in the IAM service in Oracle Cloud Infrastructure. For more information, see Managing Oracle Identity Cloud Service Users and Groups in the Oracle Cloud Infrastructure Console.

How do I find the client ID and client secret?

To edit mappings of your user groups in Oracle Identity Cloud Service to user groups in Oracle Cloud Infrastructure, you'll need to supply the client ID and client secret. The client ID and client secret are stored in Oracle Identity Cloud Service. To get this information:

1. Sign in to the Oracle Identity Cloud Service console through My Services.
2. In the Identity Cloud Service console, click **Applications**. The list of trusted applications is displayed.
3. Click COMPUTEBAREMETAL.
4. Click **Configuration**.
5. Expand **General Information**. The client ID is displayed. Click **Show Secret** to

display the client secret.



How do I sign out from a single sign-on session?

When you click **Sign Out** from the Oracle Cloud Infrastructure Console, you are not signed out of your federated identity provider (Oracle Identity Cloud Service, in this case). To sign out of Oracle Identity Cloud Service, you need to go to your My Services console or to the Oracle Identity Cloud Service console and click Sign Out from there

If I delete the federation, can I later recreate it?

Yes. To recreate the federation with Oracle Identity Cloud Service, follow the instructions in

the topic Federating with Oracle Identity Cloud Service.

Federating with Microsoft Active Directory

This topic describes how to federate with Microsoft Active Directory using Microsoft Active Federation Services (AD FS).



Note

Before following the steps in this topic, see Federating with Identity Providers to ensure that you understand general federation concepts.

About Federating with Microsoft Active Directory

Your organization can have multiple Active Directory accounts (e.g., one for each division of the organization). You can federate multiple Active Directory accounts with Oracle Cloud Infrastructure, but each federation trust that you set up must be for a single Active Directory account.

To federate with Active Directory, you set up a trust between Active Directory and Oracle Cloud Infrastructure. To set up this trust, you perform some steps in the Oracle Cloud Infrastructure Console and some steps in Active Directory Federation Services.

Following is the general process an administrator goes through to set up federation with Active Directory. Details for each step are given in the sections below.

1. Get required information from Active Directory Federation Services.
2. Federate Active Directory with Oracle Cloud Infrastructure:
    a. Add the identity provider (AD FS) to your tenancy and provide the required information.
    b. Map Active Directory groups to IAM groups.

3. In Active Directory Federation Services, add Oracle Cloud Infrastructure as a trusted, relying party.
4. In Active Directory Federation Services, add the claim rules required in the authentication response by Oracle Cloud Infrastructure.
5. Test your configuration by logging in to Oracle Cloud Infrastructure with your Active Directory credentials.

Federating with Active Directory

Prerequisites

You have installed and configured Microsoft Active Directory Federation Services for your organization.

You have set up groups in Active Directory to map to groups in Oracle Cloud Infrastructure.



Tip

Consider naming Active Directory groups that you intend to map to Oracle Cloud Infrastructure groups with a common prefix, to make it easy to apply a filter rule. For example, OCI_Administrators, OCI_NetworkAdmins, OCI_InstanceLaunchers.

Step 1: Get required information from Active Directory Federation Services

Summary: Get the SAML metadata document and the names of the Active Directory groups that you want to map to Oracle Cloud Infrastructure Identity and Access Management groups.

1. Locate the SAML metadata document for your AD FS federation server. By default, it is located at this URL:

```
https://<yourservername>/FederationMetadata/2007-06/FederationMetadata.xml
```

Download this document and make a note of where you save it. You will upload this document to the Console in the next step.

2. Note all the Active Directory groups that you want to map to Oracle Cloud Infrastructure IAM groups. You will need to enter these in the Console in the next step.

Step 2: Add Active Directory as an identity provider in Oracle Cloud Infrastructure

Summary: Add the identity provider to your tenancy. You can set up the group mappings at the same time, or set them up later.

1. Go to the Console and sign in with your Oracle Cloud Infrastructure login and password.
2. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
3. Click **Add identity provider**.
4. Enter the following:
    a. **Display Name:** A unique name for this federation trust. This is the name federated users see when choosing which identity provider to use when signing in to the Console (e.g., *ABCCorp_ADFS* shown in the screenshot in Experience for Federated Users). The name must be unique across all identity providers you add to the tenancy. You cannot change this later.
    b. **Description:** A friendly description.
    c. **Type:** Select Microsoft Active Directory Federation Services (ADFS) or SAML 2.0 compliant identity provider.
    d. **XML:** Upload the FederationMetadata.xml file you downloaded in Step 1.
    e. Click **Show Advanced Options**.
    f. **Encrypt Assertion:** Select the check box. Microsoft AD FS encrypts the assertion by default. Selecting the check box lets the IAM service know to expect the encryption from AD FS. For more information, see Encrypt Assertion.
    g. Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information

about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

5. Click **Continue**.
6. Set up the mappings between Active Directory groups and IAM groups in Oracle Cloud Infrastructure. A given Active Directory group can be mapped to zero, one, or multiple IAM groups, and vice versa. However, each individual mapping is between only a single Active Directory group and a single IAM group. Changes to group mappings take effect typically within seconds in your home region, but may take several minutes to propagate to all regions.



Note

If you don't want to set up the group mappings now, you can simply click **Create** and come back to add the mappings later.

To create a group mapping:

a. Under **Identity Provider Group**, enter the Active Directory group name. You must enter the name exactly, including the correct case.
Select the IAM group you want to map from the list under **Oracle Cloud Infrastructure Group**. If you instead want to create a new IAM group, select **New OCI Group** and enter the name of the new group in **New OCI Group Name**. The new group is automatically created in IAM and mapped to the IdP group. It will also automatically be given this description, which you can't change: "Group created during federation".

> **Tip**
>
> Requirements for IAM group name: No spaces. Allowed characters: letters, numerals, hyphens, periods, underscores, and plus signs (+). The name cannot be changed later.

b. Repeat the above sub-steps for each mapping you want to create, and then click **Create**.

The identity provider is now added to your tenancy and appears in the list on the **Federation** page. Click the identity provider to view its details and the group mappings you just set up.

Oracle assigns the identity provider and each group mapping a unique ID called an Oracle Cloud ID (OCID). For more information, see Resource Identifiers.

In the future, come to the **Federation** page if you want to edit the group mappings or delete the identity provider from your tenancy.

Step 3: Copy the URL for the Oracle Cloud Infrastructure Federation Metadata document

Summary: The Federation page displays a link to the Oracle Cloud Infrastructure Federation Metadata document. Before you move on to configuring Active Directory Federation Services, you need to copy the URL.

1. On the Federation page, click **Download this document**.
2. Copy the URL. The URL looks similar to:

```
https://auth.r2.oracleiaas.com/v1/saml/ocid1.tenancy.oc1..aaaaaaaaqdt2tvdmhsa3jmvc5dzulgs3pcv6imfwfgdya4aq/metadata.xml
```

Step 4: In Active Directory Federation Services, add Oracle Cloud Infrastructure as a trusted relying party

1. Go to the AD FS Management Console and sign in to the account you want to federate.
2. Add Oracle Cloud Infrastructure as a **trusted relying party**:

a. From the AD FS Management Console, right-click AD FS and select **Add Relying Party Trust**.
b. In the **Add Relying Party Trust Wizard**, click **Start**.
c. Select **Import data about the relying party published online or on a local network**.

   Paste the Oracle Cloud Infrastructure Federation Metadata URL that you copied in Step 3. Click **Next**.

   AD FS will connect to the URL. If you get an error during the attempt to read the federation metadata, you can alternatively upload the Oracle Cloud Infrastructure Federation Metadata XML document.

   To upload the federation metadata document

   i. In a web browser, paste the Oracle Cloud Infrastructure Federation Metadata URL in the address bar.
   ii. Save the XML document to a location that is accessible by your AD FS Management Console.
   iii. In the **Select Data Source** step of the **Add Relying Party Trust Wizard**, select **Import data about the relying party from a file**.
   iv. Click **Browse** and select the metadata.xml file that you saved.
   v. Click **Next**.

d. Set the display name for the relying party (e.g., Oracle Cloud Infrastructure) and then click **Next**.
e. Select **I do not want to configure multi-factor authentication settings for this relying party trust at this time**.
f. Choose the appropriate Issuance Authorization Rules to either permit or deny all users access to the relying party. Note that if you choose "Deny", then you must later add the authorization rules to enable access for the appropriate users.

   Click **Next**.

g. Review the settings and click **Next**.

h. Check **Open the Edit Claim Rules** dialog for this relying part trust when the wizard closes and then click **Close**.

STEP 5: ADD THE CLAIM RULES FOR THE ORACLE CLOUD INFRASTRUCTURE RELYING PARTY

Summary: Add the claim rules so that the elements that Oracle Cloud Infrastructure requires (Name ID and groups) are added to the SAML authentication response.

Add the Name ID rule:

1. In the **Add Transform Claim Rule Wizard**, select **Transform an Incoming Claim**, and click **Next**.
2. Enter the following:
   - **Claim rule name:** Enter a name for this rule, e.g., nameid.
   - **Incoming claim type:** Select Windows account name.
   - **Outgoing claim type:** Select Name ID.
   - **Outgoing name ID format:** Select Persistent Identifier.
   - Select **Pass through all claim value**.
   - Click **Finish**.
3. The rule is displayed in the rules list. Click **Add Rule**.

Add the groups rule:



Important

Any users who are in more than 100 IdP groups cannot be authenticated to use the Oracle Cloud InfrastructureConsole. To enable authentication, apply a filter to the groups rule, as described below.

If your Active Directory users are in no more than 100 groups

Add the groups rule:

1. Under Claim rule template, select **Send Claims Using a Custom Rule**. Click **Next**.
2. In the **Add Transform Claim Rule Wizard**, enter the following:
    a. **Claim rule name:** Enter groups.
    b. **Custom rule:** Enter the following custom rule:

```
c:[Type == "http://schemas.microsoft.com/ws/2008/06/identity/claims/windowsaccountname",
Issuer == "AD AUTHORITY"] => issue(store = "Active Directory", types =
("https://auth.oraclecloud.com/saml/claims/groupName"), query = ";tokenGroups;{0}", param
= c.Value);
```

    c. Click **Finish**.

If your Active Directory users are in more than 100 groups

Add the groups rule with a filter:

To limit the groups sent to Oracle Cloud Infrastructure, create two custom claim rules. The first one retrieves all groups the user belongs to directly and indirectly. The second rule applies a filter to limit the groups passed to the service provider to only those that match the filter criteria.

Add the first rule:

1. In the Edit Claim Rules dialog, click **Add Rule**.
2. Under Claim rule template, select **Send Claims Using a Custom Rule**. Click **Next**.
3. In the **Add Transform Claim Rule Wizard**, enter the following:
    a. **Claim rule name:** Enter a name, for example, groups.
    b. **Custom rule:** Enter the following custom rule:

```
c:[Type == "http://schemas.microsoft.com/ws/2008/06/identity/claims/windowsaccountname",
Issuer == "AD AUTHORITY"] => add(store = "Active Directory", types =
```

```
("https://auth.oraclecloud.com/saml/claims/groupName"), query = ";tokenGroups;{0}", param
= c.Value);
```

Note that in this custom rule you use `add` instead of `issue`. This command passes the results of the rule to the next rule, instead of sending the results to the service provider.

c. Click **Finish**.

4. Now add the filter rule.
   a. In the Edit Claim Rules dialog, click **Add Rule**.
   b. Under Claim rule template, select **Send Claims Using a Custom Rule**. Click **Next**.
   c. In the **Add Transform Claim Rule Wizard**, enter the following:
      i. **Claim rule name:** Enter groups.
      ii. **Custom rule:** Enter an appropriate filter rule. For example to send only groups that begin with the string "OCI", enter the following:

```
c:[Type == "https://auth.oraclecloud.com/saml/claims/groupName", Value =~ "(?i)OCI"]
=> issue(claim = c);
```

This rule filters the list from the first rule to only those groups that begin with the string `OCI`. The `issue` command, sends the results of the rule to the service provider.

You can create filters with the appropriate criteria for your organization.

For information on AD FS syntax for custom rules, see the Microsoft document: Understanding Claim Rule Language in AD FS 2.0 and Higher.

      iii. Click **Finish**.

STEP 6: SET UP IAM POLICIES FOR THE GROUPS

If you haven't already, set up IAM policies to control the access the federated users have to your organization's Oracle Cloud Infrastructure resources. For more information, see Getting Started with Policies and Common Policies.

STEP 7: GIVE YOUR FEDERATED USERS THE NAME OF THE TENANT AND URL TO SIGN IN

The federated users need the URL for the Oracle Cloud Infrastructure Console (for example, https://console.us-ashburn-1.oraclecloud.com) and the name of your tenant. They'll be prompted to provide the tenant name when they sign in to the Console.

Managing Identity Providers in the Console

To add an identity provider

See Federating with Microsoft Active Directory.

To delete an identity provider

All the group mappings for the identity provider will also be deleted.

1. Delete the identity provider from your tenancy:
    a. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
    A list of the identity providers in your tenancy is displayed.
    b. Click the identity provider to view its details.
    c. Click **Delete**.
    d. Confirm when prompted.

To add group mappings for an identity provider

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
    A list of the identity providers in your tenancy is displayed.
2. Click the identity provider to view its details.
3. Click **Edit Mapping**.

4. Add at least one mapping:
    a. Click **+ Add Mapping**.
    b. Under **Identity Provider Group**, enter the Active Directory group name. The name you enter here must match exactly the name in Active Directory.
    c. Select the IAM group you want to map from the list under **Oracle Cloud Infrastructure Group**. If you instead want to create a new IAM group, select **New OCI Group** and enter the name of the new group in **New OCI Group Name**. The new group is automatically created in IAM and mapped to the IdP group. It will also automatically be given this description, which you can't change: "Group created during federation".
    d. Repeat the above sub-steps for each mapping you want to create, and then click **Submit**.

Your changes take effect typically within seconds.

To update a group mapping

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
   A list of the identity providers in your tenancy is displayed.
2. Click the identity provider to view its details.
3. Click **Edit Mapping**.
4. Update the mappings (or click the X to delete a mapping), and then click **Submit**.

Your changes take effect typically within seconds.

To delete a group mapping

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.

A list of the identity providers in your tenancy is displayed.

2. Click the identity provider to view its details.
3. For the mapping you want to delete, click **Delete** next to it.
4. Confirm when prompted.

Your changes take effect typically within seconds.

Managing Identity Providers in the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations:

Identity providers:

- CreateIdentityProvider
- ListIdentityProviders
- GetIdentityProvider
- UpdateIdentityProvider
- DeleteIdentityProvider: Before you can use this operation, you must first use DeleteIdpGroupMapping to remove all the group mappings for the identity provider.

Group mappings:

- CreateIdpGroupMapping: Each group mapping is a separate entity with its own OCID.
- ListIdpGroupMappings
- GetIdpGroupMapping
- UpdateIdpGroupMapping
- DeleteIdpGroupMapping

Federating with SAML 2.0 Identity Providers

This topic describes the general steps to federate Oracle Cloud Infrastructure with any identity provider that supports the Security Assertion Markup Language (SAML) 2.0 protocol. If you want specific instructions for Oracle Identity Cloud Service or Microsoft Active Directory, see Federating with Oracle Identity Cloud Service or Federating with Microsoft Active Directory.



Tip

Find detailed setup steps for more IdPs in the following white papers:

- Oracle Cloud Infrastructure Okta Configuration for Federation and Provisioning
- Federating Oracle Access Manager to Oracle Cloud Infrastructure

Instructions for Federating

Following is the general process an administrator goes through to set up the identity provider, and below are instructions for each step. It's assumed that the administrator is an Oracle Cloud Infrastructure user with the required credentials and access.



Note

Before following the steps in this topic, see Federating with Identity Providers to ensure that you understand general federation concepts.

1. In the Oracle Cloud Infrastructure Console, get the federation metadata required to establish a trust relationship with the Identity Provider (IdP).

2. In the IdP, configure Oracle Cloud Infrastructure as an application (sometimes called a trusted relying party).
3. In the IdP, assign users and groups to your new Oracle Cloud Infrastructure application.
4. In the IdP, get the required information needed by Oracle Cloud Infrastructure.
5. In Oracle Cloud Infrastructure:
    a. Add the identity provider to your tenancy and provide information you got from the IdP.
    b. Map the IdP's groups to IAM groups.
6. In Oracle Cloud Infrastructure, make sure you have IAM policies set up for the groups so you can control users' access to Oracle Cloud Infrastructure resources.
7. Inform your users of the name of your Oracle Cloud Infrastructure tenant and the URL for the Console (for example, https://console.us-ashburn-1.oraclecloud.com).

Step 1: Get information from Oracle Cloud Infrastructure

Summary: Download the federation metadata document.

The federation metadata document is a standard SAML 2.0 document, which provides information about Oracle Cloud Infrastructure you'll need to provide to your IdP. Depending on your provider's setup requirements, you may need to upload the entire document, or you may be asked to provide only specific metadata values from the document.

1. Sign in to the Oracle Cloud Infrastructure Console as an administrator.
2. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
3. Right-click the **Download this document** link and save the document.

Step 2: Set up Oracle Cloud Infrastructure as a trusted application

Consult your IdP documentation for how to set up a trusted application. Refer to the metadata document you downloaded for required parameters.

Step 3: Assign users and groups to the new application.

Follow your IdP's procedures for adding users and groups to the application you set up for Oracle Cloud Infrastructure.

Step 4: Download the IdP's metadata document.

Your IdP should provide a SAML 2.0 document that contains the information Oracle Cloud Infrastructure needs to complete the federation. See your IdP documentation for instructions on downloading this document.

Step 5: Federate the IdP with Oracle Cloud Infrastructure

Summary: Add the identity provider to your tenancy. You can set up the group mappings at the same time, or set them up later.

Details:

1. Go to the Console and sign in with your Oracle Cloud Infrastructure login and password.
2. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
3. Click **Add Identity Provider**.
4. Enter the following:
    a. **Name:** A unique name for this federation trust. This is the name federated users see when choosing which identity provider to use when signing in to the Console, so consider making this a friendly, intuitive name your users will understand. The name must be unique across all identity providers you add to the tenancy. You cannot change this later.
    b. **Description:** A friendly description.
    c. **Type:** Select **Microsoft Active Directory Federation Service (ADFS) or SAML 2.0 Compliant Identity Provider**.
    d. **XML:** Upload the metadata.xml document that you downloaded from your IdP.
    e. Click **Show Advanced Options**.

f. **Encrypt Assertion:** Selecting the check box lets the IAM service know to expect the encryption from the IdP. If you select this check box, you must also set up encryption of the assertion in your IdP. For more information, see Encrypt Assertion. See also your IdP's documentation.

g. Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

5. Click **Continue**.

6. Set up the mappings between the IdP groups and IAM groups in Oracle Cloud Infrastructure. A given IdP group can be mapped to zero, one, or multiple IAM groups, and vice versa. However, each individual mapping is between only a single IdP group and a single IAM group. Changes to group mappings take effect typically within seconds in your home region, but may take several minutes to propagate to all regions.



Note

If you don't want to set up the group mappings now, you can simply click **Create** and come back to add the mappings later.

To create a group mapping:

a. Under **Identity Provider Group**, enter the name of the group in your IdP. You must enter the name exactly, including the correct case.

Select the IAM group you want to map from the list under **Oracle Cloud Infrastructure Group**. If you instead want to create a new IAM group, select **New OCI Group** and enter the name of the new group in **New OCI Group Name**. The new group is automatically created in IAM and mapped to the IdP

group. It will also automatically be given this description, which you can't change: "Group created during federation".



Tip

Requirements for IAM group name: No spaces. Allowed characters: letters, numerals, hyphens, periods, underscores, and plus signs (+). The name cannot be changed later.

b. Repeat the above sub-steps for each mapping you want to create, and then click **Create**.

The identity provider is now added to your tenancy and appears in the list on the **Federation** page. Click the identity provider to view its details and the group mappings you just set up.

Oracle assigns the identity provider and each group mapping a unique ID called an Oracle Cloud ID (OCID). For more information, see Resource Identifiers.

In the future, come to the **Federation** page if you want to edit or add group mappings or delete the identity provider from your tenancy.

Step 6: Set up IAM policies for the groups

If you haven't already, set up IAM policies to control the access the federated users have to your organization's Oracle Cloud Infrastructure resources. For more information, see Getting Started with Policies and Common Policies.

Step 7: Give your federated users the name of the tenant and URL to sign in

The federated users need the URL for the Oracle Cloud Infrastructure Console (for example, https://console.us-ashburn-1.oraclecloud.com) and the name of your tenant. They'll be prompted to provide the tenant name when they sign in to the Console.

Managing Identity Providers in the Console

To add an identity provider

See Instructions for Federating.

To delete an identity provider

All the group mappings for the identity provider will also be deleted.

1. Delete the identity provider from your tenancy:
    a. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
    A list of the identity providers in your tenancy is displayed.
    b. Click the identity provider to view its details.
    c. Click **Delete**.
    d. Confirm when prompted.
2. Follow your IdP's documentation to delete the application from your IdP.

To add group mappings for an identity provider

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
    A list of the identity providers in your tenancy is displayed.
2. Click the identity provider to view its details.
3. Click **Edit Mapping**.
4. Add at least one mapping:
    a. Click **+ Add Mapping**.
    b. Enter the IdP group name exactly in the **Identity Provider Group** text box.

c. Select the IAM group you want to map from the list under **Oracle Cloud Infrastructure Group**. If you instead want to create a new IAM group, select **New OCI Group** and enter the name of the new group in **New OCI Group Name**. The new group is automatically created in IAM and mapped to the IdP group. It will also automatically be given this description, which you can't change: "Group created during federation".

d. Repeat the above sub-steps for each mapping you want to create, and then click **Submit**.

Your changes take effect typically within seconds in your home region. Wait several more minutes for changes to propagate to all regions

To update a group mapping

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
   A list of the identity providers in your tenancy is displayed.
2. Click the identity provider to view its details.
3. Click **Edit Mapping**.
4. When prompted, provide the client ID and client secret for the Oracle Identity Cloud Service application, and then click **Continue**.
5. Update the mappings (or click the X to delete a mapping), and then click **Submit**.

Your changes take effect typically within seconds in your home region. Wait several more minutes for changes to propagate to all regions

To delete a group mapping

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
   A list of the identity providers in your tenancy is displayed.

2. Click the identity provider to view its details.
3. For the mapping you want to delete, click **Delete** next to it.
4. Confirm when prompted.

Your changes take effect typically within seconds in your home region. Wait several more minutes for changes to propagate to all regions.

Managing Identity Providers in the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations:

Identity providers:

- CreateIdentityProvider
- ListIdentityProviders
- GetIdentityProvider
- UpdateIdentityProvider
- DeleteIdentityProvider: Before you can use this operation, you must first use DeleteIdpGroupMapping to remove all the group mappings for the identity provider.

Group mappings:

- CreateIdpGroupMapping: Each group mapping is a separate entity with its own OCID.
- ListIdpGroupMappings
- GetIdpGroupMapping
- UpdateIdpGroupMapping
- DeleteIdpGroupMapping

User Provisioning for Federated Users

This topic describes how you can use SCIM to provision federated users in Oracle Cloud Infrastructure. Provisioned federated users can have API keys and other service-specific credentials.

Overview

SCIM (System for Cross-domain Identity Management) is an IETF standard protocol that enables user provisioning across identity systems. Oracle Cloud Infrastructure hosts a SCIM endpoint for provisioning federated users into Oracle Cloud Infrastructure. If your IdP is Oracle Identity Cloud Service or Okta, you can set up SCIM user provisioning.

After you configure the SCIM integration between your IdP and Oracle Cloud Infrastructure, users that belong to groups mapped to Oracle Cloud Infrastructure groups are automatically provisioned in Oracle Cloud Infrastructure. Provisioned users are assigned a unique OCID, and can have API keys and other service-specific credentials.

The following functionality is supported for provisioned, federated users:

- Provisioned users are assigned a unique OCID
- Provisioned users can have API keys, auth tokens, and other service-specific credentials
- You can list the users in the Console
- Provisioned users can access the **User Settings** page to see and manage these credentials for themselves
- When you add or remove users to Oracle Cloud Infrastructure-mapped groups in your IdP, the updates are automatically synched with Oracle Cloud Infrastructure

Understanding User Types

The SCIM configuration introduces the concept of the *provisioned* or *synchronized* user. The following descriptions provide details to help you understand the user types you'll be managing.

- Federated users

  A federated user is created and managed in an identity provider. Federated users can sign in to the Console using a password managed in their identity provider. Federated users are granted access to Oracle Cloud Infrastructure based on their membership in groups that are mapped to Oracle Cloud Infrastructure groups.

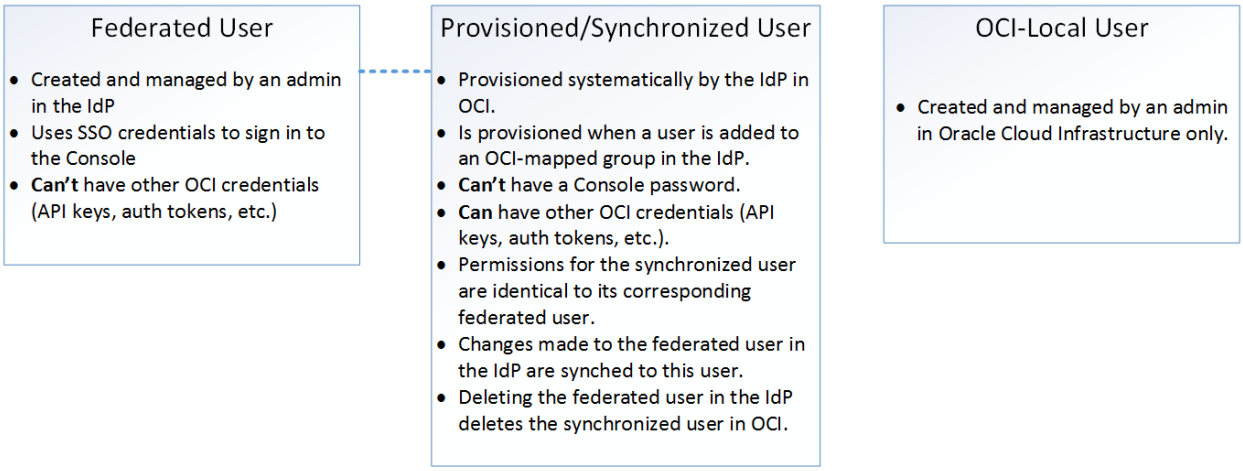
- Provisioned (or Synchronized) users

  A synchronized user is systematically provisioned by the identity provider in Oracle Cloud Infrastructure. Synchronized users can have Oracle Cloud Infrastructure credentials, but not Console passwords. When listing users in the Console, you can identify synchronized users using the **User Type** filter.

- Local users

  A local user is a user created and managed in Oracle Cloud Infrastructure's IAM service. Federated tenancies typically would have few, if any, local users. When listing users in the Console, you can identify local users using the **User Type** filter.

The following graphic summarizes the characteristics of the user types:



Who Should Set Up This Integration?

Set up this integration if your IdP is Oracle Identity Cloud Service or Okta and your federated users need to have the specialized credentials required by some services and features. For

example, if you need your federated users to access Oracle Cloud Infrastructure through the SDK or CLI, setting up this integration enables these users to get the API keys needed for this access.

Prerequisite

Perform this synchronization setup after you have successfully set up a federation between your IdP and Oracle Cloud Infrastructure. See Supported Identity Providers.

Enabling User Provisioning

Instructions for Oracle Identity Cloud Service Federations

If your identity provider is Oracle Identity Cloud Service, you need to perform a one-time upgrade.



Important

If your tenancy was created December 21, 2018 or later, your tenancy is automatically configured to provision your Oracle Identity Cloud Service users in Oracle Cloud Infrastructure. You do not need to perform the steps in this topic. See Understanding User Types and Managing User Capabilities for Federated Users for information on managing your federated users.

Upgrading Your Oracle Identity Cloud Service Federation

If your federation with Oracle Identity Cloud service was set up before December 21, 2018, perform this one-time upgrade task.

To upgrade your Oracle Identity Cloud Service federation:

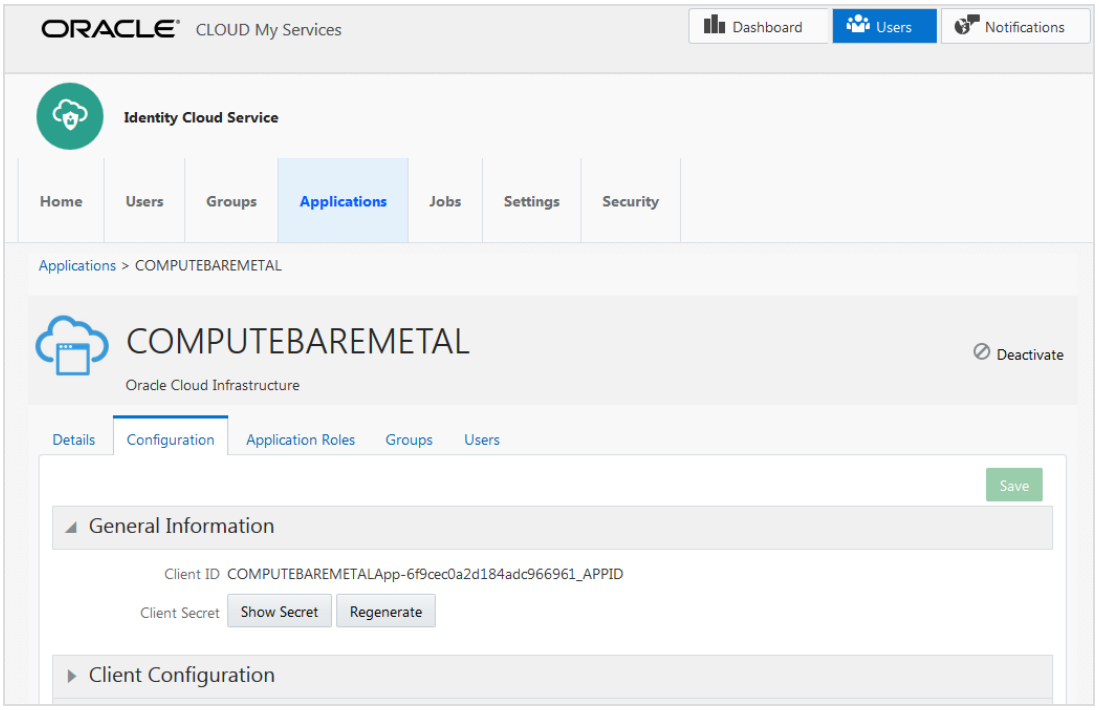
1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Federation**.
   A list of the identity providers in your tenancy is displayed.
2. Click your Identity Cloud Service federation to view its details. If your tenancy was auto-federated, it is listed as OracleIdentityCloudService.
3. Click **Edit Mapping**.
4. When prompted, provide the client ID and client secret for the Oracle Identity Cloud Service application, and then click **Continue**.

Where do I find the client ID and client secret?

The client ID and client secret are stored in Oracle Identity Cloud Service. To get this information:

a. Sign in to the Oracle Identity Cloud Service console through My Services.
b. In the Identity Cloud Service console, click **Applications**. The list of trusted applications is displayed.
c. Click COMPUTEBAREMETAL.
d. Click **Configuration**.
e. Expand **General Information**. The client ID is displayed. Click **Show Secret** to

display the client secret.



Allow several minutes for the changes to take effect.

Instructions for Okta Federations

If you do not have an existing federation with Okta, follow the instructions in the white paper, Oracle Cloud Infrastructure Okta Configuration for Federation and Provisioning. This paper includes instructions for both setting up your federation and provisioning with SCIM.

If you have an existing federation with Okta with group mappings that you want to maintain, you can add SCIM provisioning as follows:

1. In Okta, delete the existing SAML application you originally set up to federate with Oracle Cloud Infrastructure.
2. Set up a new SAML application in Okta according to the instructions in the white paper, Oracle Cloud Infrastructure Okta Configuration for Federation and Provisioning, with the following exceptions:
    - Skip the steps to **Add Identity Provider** to Oracle Cloud Infrastructure (you already have this resource in Oracle Cloud Infrastructure).
    - Instead, click **Edit Identity Provider** and upload the new metadata.xml document from the new Okta app you created.
    - Then, in Oracle Cloud Infrastructure, ensure that you **Reset Credentials**. Add the new Client ID and Secret to the API integration settings page in Okta (Step 7 in the white paper).

What to Expect After the Upgrade

When the system has had time to synchronize, you can manage user capabilities for federated users in the Console. Users that belong to a group mapped to a group in Oracle Cloud Infrastructure are listed on the Users page in the Console. Whenever you add new users to mapped groups in Oracle Identity Cloud Service, they will be available in the Console after the system synchronizes.

By default, the following user capabilities are enabled:

- API keys
- auth tokens
- SMTP credentials
- customer secret keys

Notice that you can't enable a local password. The Oracle Cloud Infrastructure console password is still managed only in your IdP.

For more information about user capabilities, see Managing User Capabilities for Federated Users.

Resetting Credentials

Use the **Reset Credentials** button to reset your SCIM client credentials. You can perform this task periodically as a security measure to rotate your credentials. After you reset these credentials, you'll need to update the SAML app in your identity provider with the new credentials.

Note: If your IdP is Oracle Identity Cloud Service, Oracle Cloud Infrastructure automatically resets the credentials with Oracle Identity Cloud Service for you. You don't need to manually reset the configuration.

Actions You Still Perform in Your Identity Provider

After the integration is set up, continue to perform the following actions in your IdP:

- Create users and assign them to groups.
- Delete users.
  Users that you delete from your IdP are removed from Oracle Cloud Infrastructure when the next synching cycle completes.
- Query for group membership.
- Manage sign-in passwords for users.

Managing User Capabilities for Federated Users

This topic describes managing user capabilities for federated users when your tenancy is federated and configured for user provisioning with a supported identity provider.

About User Capabilities

To access Oracle Cloud Infrastructure, a user must have the required credentials. Users who need to use the Console, must have a password. Users who need access through the API need API keys. Some service features require additional credentials, such as auth tokens, SMTP credentials, and Amazon S3 Compatibility API keys. For a user to get these credentials, the user must be granted the capability to have the credential type.

User capabilities are managed by an Administrator in the user's details. Each user can see their capabilities, but only an Administrator can enable or disable them. The user capabilities available to federated users are:

- API keys
- auth tokens
- SMTP credentials
- customer secret keys

By default, these capabilities are enabled when you provision new users, allowing users to create these credentials for themselves. For information about these user credentials, see Managing User Credentials.



Important

The capability "Console password" is not available for federated users. Federated users authenticate to the Console through their IdP, where their sign-in passwords are managed.

Required IAM Policy

If you're in the Administrators group, then you have the required access for managing user capabilities. A user can't enable or disable user capabilities for themselves (except for Administrators). However, a user can manage their own credentials that have been enabled for them.

Prerequisites

Management of user capabilities for federated users is supported for Oracle Identity Cloud Service and Okta federations only.

- Oracle Identity Cloud Service federations:
  If your tenancy was created December 21, 2018 or later, your tenancy is automatically configured to manage user capabilities. There are no prerequisites.
  If your tenancy was created before December 21, 2018, you must perform a one-time upgrade. See Instructions for Oracle Identity Cloud Service Federations.
- If your tenancy is federated with Okta, see User Provisioning for Federated Users.

Viewing Provisioned Federated Users in the Console

After the prerequisites are satisfied, you can view users that you create in your IdP that belong to groups mapped to Oracle Cloud Infrastructure groups. Whenever you add a user to a group mapped to an Oracle Cloud Infrastructure group, the user automatically displays in the Console.

To list users in the Console:

Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Users**.

Notice that you can filter the list by user type to include only users that belong to a specified identity provider. **Local Users** are users created in Oracle Cloud Infrastructure's IAM service. The filter list includes all identity providers you have set up.

Using the Console

To edit user capabilities

If you're an Administrator, you can edit user capabilities.

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Users**.
   A list of the users in your tenancy is displayed.
2. Click the user to see its details.

3. Click **Edit User Capabilities**.
4. Select or clear the check box to add or remove a capability.
5. Click **Save**.

To change a user's description

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Users**.
   A list of the users in your tenancy is displayed.
2. Click the user you want to update.
   The user's details are displayed. The description is displayed under the user's login.
3. Click the pencil next to the description.
4. Edit the description and save it. This description is maintained in Oracle Cloud Infrastructure and is not synched back to your identity provider.

To apply tags to a user

For instructions, see Resource Tags.

To delete a user

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Users**.
   A list of the users in your tenancy is displayed.
2. Find the user you want to delete and click the Actions icon (three dots).
3. Click **Delete**.

Important: Deleting a user here does not delete the user in your IdP. If you later want the federated user to have a provisioned user in Oracle Cloud Infrastructure, you must remove the user from all OCI-mapped groups in Oracle Identity Cloud Service and re-add the user.

For information about managing user credentials in the Console, see Managing User Credentials.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage user capabilities:

- ListUsers
- GetUser
- UpdateUser: You can update the user capabilities and the user's description.
- UpdateUserCapabilities
- DeleteUser: This operation deletes the provisioned user in Oracle Cloud Infrastructure, but not the user in the identity provider.

For information about the API operations for managing user credentials, see Managing User Credentials.

The following operations are *not* supported for federated users:

- ListUserGroupMemberships
- AddUserToGroup
- GetUserGroupMembership
- RemoveUserFromGroup

Calling Services from an Instance

This topic describes how you can authorize instances to call services in Oracle Cloud Infrastructure.

Introduction

This procedure describes how you can authorize an instance to make API calls in Oracle Cloud Infrastructure services. After you set up the required resources and policies, an application running on an instance can call Oracle Cloud Infrastructure public services, removing the need to configure user credentials or a configuration file.

Concepts

DYNAMIC GROUP

Dynamic groups allow you to group Oracle Cloud Infrastructure instances as principal actors, similar to user groups. You can then create policies to permit instances in these groups to make API calls against Oracle Cloud Infrastructure services. Membership in the group is determined by a set of criteria you define, called *matching rules*.

MATCHING RULE

When you set up a dynamic group, you also define the rules for membership in the group. Resources that match the rule criteria are members of the dynamic group. Matching rules have a specific syntax you follow. See Writing Matching Rules to Define Dynamic Groups.

INSTANCE PRINCIPALS

The IAM service feature that enables instances to be authorized actors (or principals) to perform actions on service resources. Each compute instance has its own identity, and it authenticates using the certificates that are added to it. These certificates are automatically created, assigned to instances and rotated, preventing the need for you to distribute credentials to your hosts and rotate them.

Security Considerations

Any user who has access to the instance (who can SSH to the instance), automatically inherits the privileges granted to the instance. Before you grant permissions to an instance using this procedure, ensure that you know who can access it, and that they should be authorized with the permissions you are granting to the instance.

Process Overview

The following steps summarize the process flow for setting up and using instances as principals. The subsequent sections provide more details.

1. Create a dynamic group. In the dynamic group definition, you provide the matching rules to specify which instances you want to allow to make API calls against services.
2. Create a policy granting permissions to the dynamic group to access services in your tenancy (or compartment).
3. A developer in your organization configures the application built using the Oracle Cloud Infrastructure SDK to authenticate using the instance principals provider. The developer deploys the application and the SDK to all the instances that belong to the dynamic group.
4. The deployed SDK makes calls to Oracle Cloud Infrastructure APIs as allowed by the policy (without needing to configure API credentials).
5. For each API call made by an instance, the Audit service logs the event, recording the OCID of the instance as the value of `principalId` in the event log.

Steps to Enable Instances to Call Services

Perform these tasks to enable an instance to call services:

Create a Dynamic Group and Matching Rules

Write Policies for Dynamic Groups

Configure the SDK, CLI, or Terraform

Creating a Dynamic Group and Matching Rules

See Managing Dynamic Groups.

Writing Policies for Dynamic Groups

After you have created a dynamic group, you need to create policies to permit the dynamic groups to access Oracle Cloud Infrastructure services.

Policy for dynamic groups follows the syntax described in How Policies Work. Review that topic to understand basic policy features.

The syntax to permit a dynamic group access to resources in a compartment is:

```
Allow dynamic-group <dynamic_group_name> to <verb> <resource-type> in compartment <compartment_name>
```

The syntax to permit a dynamic group access to a tenancy is:

```
Allow dynamic-group <dynamic_group_name> to <verb> <resource-type> in tenancy
```

Here are a few example policies:

To allow a dynamic group (FrontEnd) to use a load balancer in a specific compartment (ProjectA):

```
Allow dynamic-group FrontEnd to use load-balancers in compartment ProjectA
```

To allow a dynamic group to launch instances in a specific compartment:

```
Allow dynamic-group FrontEnd to manage instance-family in compartment ProjectA
Allow dynamic-group FrontEnd to use volume-family in compartment ProjectA
Allow dynamic-group FrontEnd to use virtual-network-family in compartment ProjectA
```

For more sample policies, see Common Policies.

Configuring the SDK, CLI, or Terraform

For information about SDKs, see Software Development Kits and Command Line Interface.

For the SDK for Java:

In your SDK for Java, create an `InstancePrincipalsAuthenticationDetailsProvider` object. For example:

```
public static void main(String[] args) throws Exception {
   InstancePrincipalsAuthenticationDetailsProvider provider =
```

```
      InstancePrincipalsAuthenticationDetailsProvider.builder().build();

   IdentityClient identityClient = new IdentityClient(provider);

...
```

For the Python SDK:

In your Python SDK, create an `oci.auth.signers.InstancePrincipalsSecurityTokenSigner` object. For example:

```
# By default this will hit the auth service in the region returned by
http://169.254.169.254/opc/v1/instance/region on the instance.

signer = oci.auth.signers.InstancePrincipalsSecurityTokenSigner()
identity_client = oci.identity.IdentityClient(config={}, signer=signer)

...
```

To refresh the token without waiting, use the following command:

```
signer.refresh_security_token()
```

Enabling Instance Principal Authorization for the CLI

To enable instance principal authorization from the CLI, you can set the authorization option (`--auth`) for a command. For example:

```
oci os ns get --auth instance_principal
```

Alternatively, you can set the following environment variable:

```
OCI_CLI_AUTH=instance_principal
```

Note that if both are set, the value set for `--auth` takes precedence over the environment variable.

For information about using the CLI, see Getting Started with the Command Line Interface.

Enabling Instance Principal Authorization for Terraform

To enable instance principal authorization in Terraform, you can set the `auth` attribute to "InstancePrincipal" in the provider definition as shown in the following sample:

```
variable "region" {}
```

```
provider "oci" {
   auth = "InstancePrincipal"
   region = "${var.region}"

}
```

Note that when you use instance principal authorization you do not need to include the `tenancy_ocid`, `user_ocid`, `fingerprint`, and `private_key_path` attributes.

FAQs

How do I query the instance metadata service to query the certificate on the instance?

Use this curl command: `curl http://169.254.169.254/opc/v1/identity/cert.pem`

How frequently is the certificate rotated on each instance?

The certificate is rotated multiple times each day.

What happens if I try to use an expired certificate?

You will get a 401-Not Authenticated error.

Can I change the frequency at which the certificate is rotated?

No. You can't change the frequency at which the certificate is rotated. However, you can change the policy on the dynamic group. If you think an instance has been compromised, you can either change the policy on the dynamic group to revoke permissions for all members of the group, or you can remove the instance from the dynamic group. See Can I remove an instance from a dynamic group?

What happens if the certificate is rotated in the middle of a long running operation?

The token expiration is independent of the certificate expiration period. And, it also depends on the application you are interacting with. For example, if Object Storage does not have a multipart PUT operation, then it does not matter how long the operation runs.

Are the certificates accessible for all users on an instance?

Yes. Ensure that only users who should be granted the access that you have granted to the dynamic group, have access to the instance.

Are dynamic groups created at the tenancy level?

Yes.

Can I remove an instance from a dynamic group?

Yes. You can remove it by modifying the matching rule to exclude it. See below for an example.

Can I exclude specific instances in a compartment from the dynamic group?

Yes. For example, assume you want to exclude two specific instances in a compartment from the dynamic group. Write a matching rule like this:

```
All {instance.compartment.id = '<compartment_ocid>',
 instance.id != '<instance1_to_exclude_ocid>', instance.id != '<instance2_to_exclude_ocid>'}
```

The above rule includes all instances in the compartment except those with the OCIDs specified.

Managing Users

This topic describes the basics of working with users.



Important

If your tenancy is federated with Oracle Identity Cloud Service, see Managing Oracle Identity Cloud Service Users and Groups in the Oracle Cloud Infrastructure Console to manage users.

Required IAM Policy

If you're in the Administrators group, then you have the required access for managing users.

You can create a policy that gives someone power to create new users and credentials, but not control which groups those users are in. See Let the Help Desk manage users.

For the reverse: You can create a policy that gives someone power to determine what groups users are in, but not create or delete users. See Let group admins manage group membership.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for users or other IAM components, see Details for IAM.

Tagging Resources

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

Working with Users

When creating a user, you must provide a unique, unchangeable *name* for the user. The name must be unique across all users within your tenancy. It will be the user's login to the Console. You might want to use a name that's already in use by your company's own identity system (e.g., Active Directory, LDAP, etc.). You must also provide the user with a *description* (although it can be an empty string), which is a non-unique, changeable description for the user. This could be the user's full name, a nickname, or other descriptive information. Oracle will also assign the user a unique ID called an Oracle Cloud ID (OCID). For more information, see Resource Identifiers.



Note

If you delete a user and then create a new user with the same name, they'll be considered different users because they'll have different OCIDs.

Oracle recommends that you supply a password recovery email address for the user. If the user forgets their password, they can request to have a temporary password sent to them using the **Forgot Password** link on the sign-on page. If no email address is present for the user, an administrator must intervene to reset their password.

A new user has no permissions until you place the user in one or more groups, and there's at least one policy that gives that group permission to either the tenancy or a compartment. Exception: each user can manage *their own* credentials they have been enabled to have. An administrator does not need to create a policy to give a user that ability. For more information, see User Credentials.



Important

After creating a new user and putting them in a group, make sure to let them know which compartment(s) they have access to.

You also need to give the new user some credentials so they can access Oracle Cloud Infrastructure. A user can have one or both of the following credentials, depending on the type of access they need: A password for using the Console, and an API signing key for using the API.

About User Capabilities

To access Oracle Cloud Infrastructure, a user must have the required credentials. Users who need to use the Console, must have a password. Users who need access through the API need API keys. Some service features require additional credentials, such as auth tokens, SMTP credentials, and Amazon S3 Compatibility API keys. For a user to get these credentials, the user must be granted the capability to have the credential type.

User capabilities are managed by an Administrator in the User details. Each user can see their capabilities, but only an Administrator can enable or disable them. The user capabilities are:

- Can use Console password (native users only)
- Can use API keys
- Can use auth tokens
- Can use SMTP credentials
- Can use customer secret keys

By default, all these capabilities are enabled when you create new users, allowing users to create these credentials for themselves. For information about working with user credentials, see Managing User Credentials.

Enabling Multi-Factor Authentication for a User

See Managing Multi-Factor Authentication for details.

Unblocking a User After Unsuccessful Sign-in Attempts

If a user tries 10 times in a row to sign in to the Console unsuccessfully, they will be automatically blocked from further sign-in attempts. An administrator can unblock the user in the Console (see To unblock a user) or with the UpdateUserState API operation.

Deleting a User

You can delete a user, but only if the user is not a member of any groups.

Limits on Users

For information about the number of users you can have, see Service Limits.

Using the Console

To create a user

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Users**.
   A list of the users in your tenancy is displayed.
2. Click **Create User**.
3. Enter the following:
   - **Name:** A unique name or email address for the user (for tips on what value to use, see Working with Users). The name must be unique across all users in your tenancy. You cannot change this later. The name must meet the following

requirements: No spaces. Only Basic Latin letters (ASCII), numerals, hyphens, periods, underscores, +, and @.

- **Description:** This could be the user's full name, a nickname, or other descriptive information. You can change this later if you want to.
- **Email:** Enter an email address for the user. This email address is used for password recovery. The email address must be unique in the tenancy.
  If the user forgets their password, they can click **Forgot Password** on the sign on page, and a temporary password will be automatically generated and sent to the email address provided here. The email address can also be updated later by the user or an administrator.
- **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

4. Click **Create**.

Next, you need to give the user permissions by adding them to at least one group. You also need to give the user the credentials they need (see Managing User Credentials).

To add a user to a group

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Users**.
   A list of the users in your tenancy is displayed.
2. Locate the user in the list.
3. Click the user.
   Its details are displayed.
4. Click **Groups**.

5. Click **Add User to Group**.
6. Select the group from the drop-down list, and then click **Add**.

Make sure to let the user know which compartment(s) they have access to.

To remove a user from a group

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Users**.
   A list of the users in your tenancy is displayed.
2. Locate the user in the list.
3. Click the user.
   Its details are displayed.
4. Click **Groups**.
5. Click the Actions icon (three dots), and then click **Remove**.
6. Confirm when prompted.

To delete a user

Prerequisite: To delete a user, the user must not be in any groups.

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Users**.
   A list of the users in your tenancy is displayed.
2. For the user you want to delete, click **Delete**.
3. Confirm when prompted.

To unblock a user

If you're an administrator, you can use the following procedure to unblock a user who has

tried 10 times in a row to sign in to the Console unsuccessfully.

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Users**.
   A list of the users in your tenancy is displayed.
2. Click the user.
   Its details are displayed, including the current status.
3. Click **Unblock**.
4. Confirm when prompted.

To change a user's description

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Users**.
   A list of the users in your tenancy is displayed.
2. Click the user you want to update.
   The user's details are displayed. The description is displayed under the user's login.
3. Click the pencil next to the description.
4. Edit the description and save it.

To edit a user's email

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Users**.
   A list of the users in your tenancy is displayed.
2. Click the user you want to update.
   The user's details are displayed.
3. Under **User Information**, click the pencil next to **Email**.
4. Enter the email address and click the save icon. The email address must be unique in the tenancy.

To edit user capabilities

If you're an Administrator, you can edit user capabilities.

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Users**.
   A list of the users in your tenancy is displayed.
2. Click the user to see its details.
3. Click **Edit User Capabilities**.
4. Select or clear the check box to add or remove a capability.
5. Click **Save**.

To apply tags to a user

For instructions, see Resource Tags.

For information about managing user credentials in the Console, see Managing User Credentials.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.



Note

Updates Are Not Immediate Across All Regions

Your IAM resources reside in your home region. To enforce policy across all regions, the IAM service replicates your resources in each region. Whenever you create or change a policy, user, or group, the changes take effect first in the home region, and then are propagated out to your other regions. It can take several minutes for changes to take effect in all regions. For example, assume you have a group with permissions to launch instances in the tenancy. If you add UserA to this group, UserA will be able to launch instances in your home region within a minute. However, UserA will not be able to launch instances in other regions until the replication process is complete. This process can take up to several minutes. If UserA tries to launch an instance before replication is complete, they will get a not authorized error.

Use these API operations to manage users:

- CreateUser
- ListUsers
- GetUser
- UpdateUserState: Unblocks a user who has tried to sign in 10 times in a row unsuccessfully.
- UpdateUser: You can update the user's description, email, and tags.
- UpdateUserCapabilities
- DeleteUser

- ListUserGroupMemberships: Use this operation to get a list of which users are in a group, or which groups a user is in.
- AddUserToGroup: This operation results in a `UserGroupMembership` object with its own OCID.
- GetUserGroupMembership
- RemoveUserFromGroup: This operation deletes a `UserGroupMembership` object.

For information about the API operations for managing user credentials, see Managing User Credentials.

Managing Groups

This topic describes the basics of working with groups.



Important

If your tenancy is federated with Oracle Identity Cloud Service, see Managing Oracle Identity Cloud Service Users and Groups in the Oracle Cloud Infrastructure Console to manage groups.

Required IAM Policy

If you're in the Administrators group, then you have the required access for managing groups.

For a policy that only gives someone power to determine what groups users are in, see Let group admins manage group membership.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for groups or other IAM components, see Details for IAM.

Tagging Resources

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

Working with Groups

When creating a group, you must provide a unique, unchangeable *name* for the group. The name must be unique across all groups within your tenancy. You must also provide the group with a *description* (although it can be an empty string), which is a non-unique, changeable

description for the group. Oracle will also assign the group a unique ID called an Oracle Cloud ID (OCID). For more information, see Resource Identifiers.



Note

If you delete a group and then create a new group with the same name, they'll be considered different groups because they'll have different OCIDs.

A group has no permissions until you write at least one policy that gives that group permission to either the tenancy or a compartment. When writing the policy, you can specify the group by using either the unique name or the group's OCID. Per the preceding note, even if you specify the group name in the policy, IAM internally uses the OCID to determine the group. For information about writing policies, see Managing Policies.

You can delete a group, but only if the group is empty.

For information about the number of groups you can have, see Service Limits.

If you're federating with an identity provider, you'll create mappings between the identity provider's groups and your IAM groups. For more information, see Federating with Identity Providers.

Using the Console

To create a group

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Groups**.
   A list of the groups in your tenancy is displayed.
2. Click **Create Group**.

3. Enter the following:
    - **Name:** A unique name for the group. The name must be unique across all groups in your tenancy. You cannot change this later.
    - **Description:** A friendly description. You can change this later if you want to.
    - **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
4. Click **Create Group**.

Next, you might want to add users to the group, or write a policy for the group. See To create a policy.

To add a user to a group

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Groups**.
   A list of the groups in your tenancy is displayed.
2. Locate the group in the list.
3. Click the group.
   Its details are displayed
4. Click **Add User to Group**.
5. Select the user from the drop-down list, and then click **Add User**.

To remove a user from a group

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity**

and click **Groups**.
A list of the groups in your tenancy is displayed.

2. Locate the group in the list.
3. Click the group to display its details.
   A list of users in the group is displayed.
4. Locate the user in the list.
5. For the user you want to remove, click **Remove**.
6. Confirm when prompted.

To delete a group

Prerequisite: To delete a group, it must not have any users in it.

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Groups**.
   A list of the groups in your tenancy is displayed.
2. Locate the group in the list.
3. For the group you want to delete, click **Delete**.
4. Confirm when prompted.

To update a group's description

This is available only through the API. If you don't have access to the API and need to update a group's description, contact Oracle Support.

To apply tags to a group

For instructions, see Resource Tags.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.



Note

Updates Are Not Immediate Across All Regions

Your IAM resources reside in your home region. To enforce policy across all regions, the IAM service replicates your resources in each region. Whenever you create or change a policy, user, or group, the changes take effect first in the home region, and then are propagated out to your other regions. It can take several minutes for changes to take effect in all regions. For example, assume you have a group with permissions to launch instances in the tenancy. If you add UserA to this group, UserA will be able to launch instances in your home region within a minute. However, UserA will not be able to launch instances in other regions until the replication process is complete. This process can take up to several minutes. If UserA tries to launch an instance before replication is complete, they will get a not authorized error.

Use these API operations to manage groups:

- CreateGroup
- ListGroups
- GetGroup

- UpdateGroup: You can update only the group's description.
- DeleteGroup
- ListUserGroupMemberships: Use to get a list of which users are in a group, or which groups a user is in.
- AddUserToGroup: This operation results in a `UserGroupMembership` object with its own OCID.
- GetUserGroupMembership
- RemoveUserFromGroup: This operation deletes a `UserGroupMembership` object.

For API operations related to group mappings for identity providers, see Federating with Identity Providers.

Managing Dynamic Groups

This topic describes how to manage dynamic groups and define the rules to determine a dynamic group's members.

About Dynamic Groups

Dynamic groups allow you to group Oracle Cloud Infrastructure computer instances as "principal" actors (similar to user groups). You can then create policies to permit instances to make API calls against Oracle Cloud Infrastructure services. When you create a dynamic group, rather than adding members explicitly to the group, you instead define a set of *matching rules* to define the group members. For example, a rule could specify that all instances in a particular compartment are members of the dynamic group. The members can change dynamically as instances are launched and terminated in that compartment.

Required IAM Policy

If you're in the Administrators group, then you have the required access for managing dynamic groups.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for dynamic groups or other IAM components, see Details for IAM.

Tagging Resources

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

Working with Dynamic Groups

When creating a dynamic group, you must provide a unique, unchangeable *name* for the dynamic group. The name must be unique across all groups within your tenancy. You must also provide the dynamic group with a *description* (although it can be an empty string), which is a non-unique, changeable description for the group. Oracle will also assign the group a unique ID called an Oracle Cloud ID (OCID). For more information, see Resource Identifiers.



Note

If you delete a dynamic group and then create a new dynamic group with the same name, they'll be considered different groups because they'll have different OCIDs.

A dynamic group has no permissions until you write at least one policy that gives that dynamic group permission to either the tenancy or a compartment. When writing the policy, you can specify the dynamic group by using either the unique name or the dynamic group's OCID. Per the preceding note, even if you specify the dynamic group name in the policy, IAM internally uses the OCID to determine the dynamic group. For information about writing policies, see Managing Policies.

You can delete a dynamic group, but only if the group is empty.

Updating Dynamic Groups

You can update the matching rules that define the members of a dynamic group. For example, you might change a matching rule that includes all instances in a compartment to exclude a particular instance. Or, you might update a rule to include a new tag value.



Important

When you make a change to a matching rule you must allow about one hour for the updated policy to take effect. For example, if you update tags on an instance to either include or exclude that instance from a dynamic group, you must wait for that policy to take effect to include or exclude the instance.

Limits on Instances in Dynamic Groups

A single compute instance can belong to a maximum of 5 dynamic groups.

Using the Console

To create a dynamic group

1. Open the Console, click **Identity**, and then click **Dynamic Groups**.
   A list of the dynamic groups in your tenancy is displayed.
2. Click **Create Dynamic Group**.
3. Enter the following:
   - **Name:** A unique name for the group. The name must be unique across all groups in your tenancy (dynamic groups and user groups). You can't change this later.

- **Description:** A friendly description. You can't change this in the Console, but you can change it Using the API.

4. Enter the **Matching Rules**. Resources that meet the rule criteria are members of the group.
   - **Rule 1:** Enter a rule following the guidelines in Writing Matching Rules to Define Dynamic Groups. You can manually enter the rule in the text box or launch the rule builder.
   - Enter additional rules as needed. To add a rule, click **+Additional Rule**.
5. Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
6. Click **Create Dynamic Group**.
   The matching rule syntax is verified, but the OCIDs are not. Be sure that the OCIDs you enter are correct.

Next, to give the dynamic group permissions, you need to write a policy. See Writing Policies for Dynamic Groups.

To delete a dynamic group

1. Open the Console, click **Identity**, and then click **Dynamic Groups**.
   A list of the dynamic groups in your tenancy is displayed.
2. Locate the dynamic group in the list.
3. For the dynamic group you want to delete, click **Delete**.
4. Confirm when prompted.

To update a dynamic group's description

This is available only through the API. If you don't have access to the API and need to update a dynamic group's description, contact Oracle Support.

To update a dynamic group's matching rules

1. Open the Console, click **Identity**, and then click **Dynamic Groups**.
   A list of the dynamic groups in your tenancy is displayed.
2. Click the dynamic group you want to update.
   The dynamic group's details are displayed.
3. Click **Edit All Matching Rules**.
4. Edit the matching rule in the text box; or, you can use the rule builder if the change is supported by the rule builder.

Writing Matching Rules to Define Dynamic Groups

Matching rules define the resources that belong to the dynamic group. In the Console, you can either enter the rule manually in the provided text box, or you can use the rule builder. The rule builder lets you make selections and entries in a dialog, then writes the rule for you, based on your entries.

You can define the members of the dynamic group based on the following:

- compartment ID - include (or exclude) the instances that reside in that compartment based on compartment OCID
- instance ID - include (or exclude) an instance based on its instance OCID
- tag namespace and tag key - include (or exclude) instances tagged with a specific tag namespace and tag key. All tag values are included. For example, include all instances tagged the with tag namespace `department` and the tag key `operations`.

- tag namespace, tag key, and tag value - include (or exclude) instances tagged with a specific value for the tag namespace and tag key. For example include all instances tagged with the tag namespace `department` and the tag key `operations` and with the value '`45`'.

A matching rule has the following syntax:

For a single condition:

```
variable =|!= 'value'
```

For multiple conditions:

```
any|all {<condition>,<condition>,...}
```

Supported variables are:

- `instance.compartment.id` - the OCID of the compartment where the instance resides
- `instance.id` - the OCID of the instance
- `tag.<tagnamespace>.<tagkey>.value` - the tag namespace and tag key. For example, `tag.operations.department.value`.
- `tag.<tagnamespace>.<tagkey>.value='<tagvalue>'` - the tag namespace, tag key, and tag value. For example, `tag.operations.department.value='45'`

Here are some examples:

Include All Instances in a Specific Compartment in the Dynamic Group

To include all instances that are in a specific compartment, add a rule with the following syntax:

```
instance.compartment.id = '<compartment_ocid>'
```

You can add that rule either directly in the text box, or you can use the rule builder.

Example entry in text box:

```
instance.compartment.id = 'ocidv1:compartment:oc1:phx:samplecompartmentocid6q6igvfauxmima74jv'
```

All instances that currently exist or get created in the compartment (identified by the OCID) are members of this group.

Include All Instances in Any of Two or More Compartments

To include all instances that reside in any of two (or more) compartments, add a rule with the following syntax:

```
Any {instance.compartment.id = '<compartment_ocid>', instance.compartment.id = '<compartment_ocid>'}
```

You can add that rule either directly in the text box, or you can use the rule builder.

Example entry in the text box:

```
Any {instance.compartment.id = 'ocidv1:compartment:oc1:phx:samplecompartmentocid6q6igvfauxmima74jv',
instance.compartment.id = 'ocidv1:compartment:oc1:phx:samplecompartmentocidythksk89ekslsoelu2'}
```

Instances that currently exist or get created in either of the specified compartments are members of this group.

Include All Instances Tagged with a Specific Namespace and Tag Key

To include all instances that are tagged with a specific tag namespace and tag key, add a rule with the following syntax:

```
tag.<tagnamespace>.<tagkey>.value
```

All instances assigned the tagnamespace.tagkey combination are included. Note that the tag value is not evaluated, so all values are included.

Example: Assume you have a tag namespace called `department` and a tag key called `operations`. You want to include all instances that are tagged with the namespace and tag key.

Enter the following rule in the text box:

```
tag.operations.department.value
```

All instances that currently exist or get created with the tag namespace and tag key `operations.department` are members of this group.

Include All Instances In a Specific Compartment with a Specific Tag Namespace, Tag Key, and Tag Value

To include all instances in a specific compartment that are tagged with a specific tag namespace, key, and value, add a rule with the following syntax:

```
All {instance.compartment.id = '<compartment_ocid>',
tag.<tagnamespace>.<tagkey>.value='<tagvalue>'}
```

All instances that are in the identified compartment and that are assigned the tagnamespace.tagkey with the specified tag value are included.

Example: Assume you have a tag namespace called `department` and a tag key called `operations`. You want to include all instances that are tagged with the value 45, that are in a particular compartment.

Enter the following statement in the text box:

```
All
{instance.compartment.id='ocidv1:compartment:oc1:phx:oc1:phx:samplecompartmentocid6q6igvfauxmima74jv,',
tag.operations.department.value='45'}
```

Include Instances in a Specific Compartment Except Those with a Specific Tag

To include all instances in a specific compartment EXCEPT those that are tagged with a specific tag namespace, key, and value, add a rule with the following syntax:

```
All {instance.compartment.id = '<compartment_ocid>',
tag.<tagnamespace>.<tagkey>.value!= '<tagvalue>'}
```

Example: Assume you have a tag namespace called `department` and a tag key called `operations`. You want to include all instances in a specific compartment, except those that are tagged with the value 45.

Enter the following statement in the text box:

```
All
{instance.compartment.id='ocidv1:compartment:oc1:phx:oc1:phx:samplecompartmentocid6q6igvfauxmima74jv,',
tag.operations.department.value!='45'}
```

Using the Rule Builder

The rule builder is a tool available from the Console to help you write matching rules. The rule builder provides menus and text boxes for you to make entries and then writes the rule for you. The rule builder does have some limitations, so you can't use it for all cases.

LIMITATIONS OF THE RULE BUILDER

The rule builder does not support the following:

- Exclusion rules - the rule builder lets you select compartment IDs and instance IDs to include only.
- Rules based on tags - the rule builder does not allow you to select tags to include in your rule. To add a rule based on tag values, you need to enter the rule in the Rule text box using the syntax above.

LAUNCHING THE RULE BUILDER

When you click **Create Dynamic Group**, the Rule Builder is displayed in the **Create Dynamic Group** dialog.

To create a matching rule using the rule builder

1. Select **Any** or **All** from the menu.
   Any includes instances that match any of the statements in the rule.
   All includes only instances that match all of the statements in the rule.
2. Select the **Attribute** type for the statement and enter the value:
   in Compartment ID includes instances in the compartment you specify.
   with Instance ID includes instances with the OCID you specify.
3. Click **+Additional line** to add more statements to this rule.

When you add multiple statements to a rule, remember that Any includes instances that match any of the statements. If you choose All, instances must match all of the specifications in the statements to be included in the group.

EXAMPLES USING THE RULE BUILDER

Include All Instances in a Specific Compartment in the Dynamic Group

To include all instances that are in a specific compartment, using the rule builder:

- Select ALL.
- **Attribute:** Select in Compartment ID.
- **Value:** Enter `ocidv1:compartment:oc1:yourcompartmentocid`

All instances that currently exist or get created in the compartment (identified by the OCID) are members of this group.

Include All Instances in Any of Two or More Compartments

To include all instances that reside in any of two (or more) compartments using the rule builder:

1. Select ANY.
2. Enter:
    - **Attribute:** Select in Compartment ID.
    - **Value:** Enter `ocidv1:compartment:oc1:phx:samplecompartmentocid6q6igvfauxmima74jv`
3. Click **+Additional Line**. Enter the following on the second line:
    - **Attribute:** Select in Compartment ID.
    - **Value:** Enter `ocidv1:compartment:oc1:phx:samplecompartmentocidythksk89ekslsoelu2`

4. Continue adding additional lines as needed for each compartment you want to include.

Instances that currently exist or get created in any of the specified compartments are members of this group.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage dynamic groups:

- CreateDynamicGroup
- ListDynamicGroups
- GetDynamicGroup
- UpdateDynamicGroup
- DeleteDynamicGroup

Managing Compartments

This topic describes the basics of working with compartments.

Required IAM Policy

If you're in the Administrators group, then you have the required access for managing compartments.

For an additional policy related to compartment management, see Let a compartment admin manage the compartment.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for compartments or other IAM components, see Details for IAM.

Tagging Resources

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

Working with Compartments

When you first start working with Oracle Cloud Infrastructure, you need to think carefully about how you want to use compartments to organize and isolate your cloud resources. Compartments are fundamental to that process. Once you put a resource in a compartment, you can't move it, so it's important to think through your compartment design for your organization up front, before implementing anything. For more information, see "Setting Up Your Tenancy" in the *Oracle Cloud Infrastructure Getting Started Guide*.

The Console is designed to display your resources by *compartment* within the current region. When you work with your resources in the Console, you must choose which compartment to work in from a list on the page. That list is filtered to show only the compartments in the

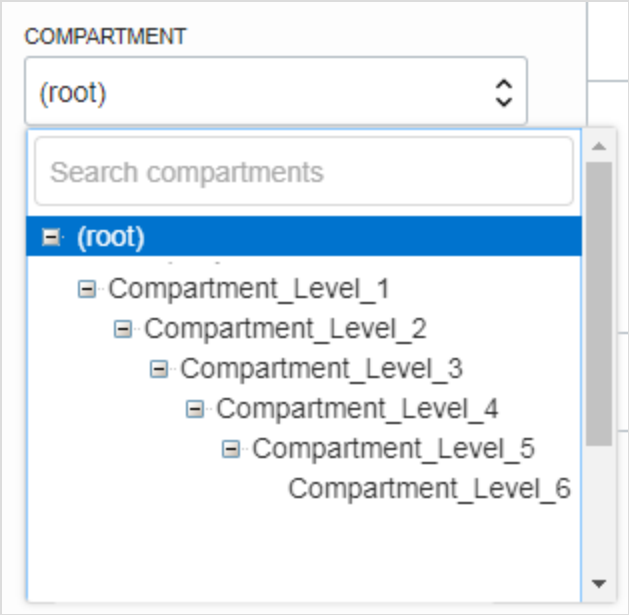
tenancy that you have permission to access. If you're an administrator, you'll have permission to view all compartments and work with any compartment's resources, but if you're a user with limited access, you probably won't.

Compartments are global, across regions. When you create a compartment, it is available in every region that your tenancy is subscribed to.

Creating Compartments

When creating a new compartment, you must provide a *name* for it (maximum 100 characters, including letters, numbers, periods, hyphens, and underscores) that is unique within its parent compartment. You must also provide a *description*, which is a non-unique, changeable description for the compartment, between 1 and 400 characters. Oracle will also assign the compartment a unique ID called an Oracle Cloud ID. For more information, see Resource Identifiers.

You can create subcompartments inside of compartments to create hierarchies that are six levels deep.



Like other resources, you can't move subcompartments from one compartment to another.

For information about the number of compartments you can have, see Service Limits.

Access Control for Compartments

After creating a compartment, you need to write at least one policy for it, otherwise no one can access it (except administrators or users who have permission to the tenancy).

When creating a compartment inside another compartment, the compartment inherits access permissions from compartments higher up its hierarchy. For more information, see Policy Inheritance.

When you create an access policy, you need to specify which compartment to attach it to. This controls who can later modify or delete the policy. Depending on how you've designed your compartment hierarchy, you might attach it to the tenancy, a parent, or to the specific compartment itself. For more information, see Policy Attachment.

Putting Resources in a Compartment

To place a new resource in a compartment, you simply specify that compartment when creating the resource (the compartment is one of the required pieces of information to create a resource). If you're working in the Console, you just make sure you're first viewing the compartment where you want to create the resource. Keep in mind that most IAM resources reside in the tenancy (this includes users, groups, compartments, and any policies attached to the tenancy). Notice that you can't move a resource from one compartment to another.

Viewing Resources in a Compartment

It's not possible to get a list of all the resources in a compartment by using a single API call. Instead you can list all the resources of a given type in the compartment (e.g., all the instances, all the block storage volumes, etc.).



Tip

Search allows you to get a list of resources in a compartment, with some limitations. Search lists the resources in the region you are viewing. Not all resources support Search. For more information, see Overview of Search.

Deleting Compartments

To delete a compartment, it must be empty of all resources. Before you initiate deleting a compartment, be sure that all its resources have been deleted or terminated, including any policies attached to the compartment.



Important

Some resource types can't be deleted, therefore, compartments containing these resource types can't be deleted. The resource types that can't be deleted are:

- Tag namespaces and tag key definitions
- Data transfer jobs

The delete action is asynchronous and initiates a work request. The state of the compartment changes to Deleting while the work request is executing. It typically takes several minutes for the work request to complete. While it is in the Deleting state it is not displayed on the compartment picker. If the work request fails, the compartment is not deleted and it returns to the Active state.

After a compartment is deleted, its state is updated to Deleted and a random string of characters is appended to its name, for example, CompartmentA might become CompartmentA.qR5hP2BD. Renaming the compartment allows you to reuse the original name

for a different compartment. The deleted compartment is displayed on the Compartments page for the number of days specified in your Audit Retention Period setting (90-365 days). The deleted compartment is removed from the compartment picker. If any policy statements reference the deleted compartment, the name in the policy statement is updated to the new name.

Troubleshooting tips for when a compartment fails to delete

If the compartment fails to delete, verify that you have removed all the resources:

- For many resources, you can use Search to help you locate them, but keep in mind that not all resources support Search. See Supported Resources for the list.

  ### To search for resources in a compartment

  1. Get the OCID of the compartment you want to search in.
  2. In the Console, append "/a/query" to the end of your base Console URL. For example, https://console.us-ashburn-1.oraclecloud.com/a/query.
  3. Open the **Select Sample Query** menu and then click **Query for all resources in a compartment**.
  4. In the query string, replace *compartmentOcid* with your compartment OCID.
  5. Click **Search**.

- Be sure to search in each region that your tenancy is subscribed to.
- Verify that there are no policies in the compartment (polices are not included in Search results).

  ### To find policies in a compartment

  1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Policies**.

2. From the compartments list on the left, select the compartment you want to delete.

   Policies attached to the compartment are displayed.

- If you can't locate any resources in the compartment, check with your Administrator; you might not have permission to view all resources.



Important

There is a known issue causing deleted compartments to continue to count against your service limit of compartments. See Deleted compartments continue to count against service limits.

Adding Tag Defaults for a Compartment

Tag defaults let you specify tags to be applied automatically to all resources, at the time of creation, in the current compartment. For more information, see Managing Tag Defaults.

Using the Console

To create a compartment

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Compartments**.
   A list of the compartments you have access to is displayed.
2. Navigate to the compartment in which you want to create the new compartment:
   - To create the compartment in the tenancy (root compartment) click **Create Compartment**.

- Otherwise, click through the hierarchy of compartments until you reach the detail page of the compartment in which you want to create the compartment. On the **Compartment Details** page, click **Create Compartment**.

3. Enter the following:
   - **Name:** A unique name for the compartment (maximum 100 characters, including letters, numbers, periods, hyphens, and underscores). The name must be unique across all the compartments in your tenancy. Avoid entering confidential information.
   - **Description:** A friendly description. You can change this later if you want to. Avoid entering confidential information.
   - **Compartment:** The compartment you are in is displayed. To choose another compartment to create this compartment in, select it from the list.
   - **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
4. Click **Create Compartment**.

Next, you might want to write a policy for the compartment. See To create a policy.

To update a compartment's name

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Compartments**.
   A list of the compartments in your tenancy is displayed.
2. For the compartment you want to rename, click the Actions icon (three dots), and then click **Rename Compartment**.



Tip

You can't change the name of your root compartment.

3. Enter the new **Name**. The name must be unique across all the compartments in your tenancy. The name can have a maximum of 100 characters, including letters, numbers, periods, hyphens, and underscores. Avoid entering confidential information.
4. Click **Rename Compartment**.

To update a compartment's description

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Compartments**.
   A list of the compartments in your tenancy is displayed.
2. For the compartment you want to update, click the Actions icon (three dots), and then click **Edit Compartment Description**.
3. Enter the new description. Avoid entering confidential information.
4. Click **Save**.

To view the contents of a compartment

1. Open the Console,
2. Open the navigation menu and select the type of resource you want to view. For example, click **Compute** to view all your Compute resources.
3. Choose the compartment from the list on the left side of the page.
   The page updates to show only the resources in that compartment.

Remember that most IAM resources reside in the tenancy (this includes users, groups, and compartments). Policies can reside in either the tenancy (root compartment) or other compartments.

To apply tags to a compartment

For instructions, see Resource Tags.

To manage tag defaults for a compartment

See Managing Tag Defaults.

To delete a compartment

You must remove all resources from a compartment before you can delete it.

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Compartments**.
   A list of the compartments in your tenancy is displayed.
2. For the compartment you want to delete, click the Actions icon (three dots), and then click **Delete Compartment**.
3. At the prompt, click **OK**.

After you click **OK**, a work request is submitted to delete the compartment. The compartment state changes to Deleting. If the work request fails, the state returns to Active.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage compartments:

- CreateCompartment
- ListCompartments
- GetCompartment: Returns the metadata for the compartment, not its contents.
- UpdateCompartment
- DeleteCompartment
- GetWorkRequest: Gets the work requests spawned by the DeleteCompartment operation.

You can retrieve the contents of a compartment only by resource type. There's no API call that lists *all* resources in the compartment. For example, to list all the instances in a compartment, call the Core Services API ListInstances operation and specify the compartment ID as a query parameter.

Managing Tags and Tag Namespaces

When you have many resources (for example, instances, VCNs, load balancers, and block volumes) across multiple compartments in your tenancy, it can become difficult to track resources used for specific purposes, or to aggregate them, report on them, or take bulk actions on them. *Tagging* allows you to define keys and values and associate them with resources. You can then use the tags to help you organize and list resources based on your business needs.

Required IAM Policy

If you're in the Administrators group, then you have the required access for managing tag namespaces and tags. For more policy samples specific to working with tags and tag namespaces, see Required Permissions for Working with Defined Tags.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for groups or other IAM components, see Details for IAM.

Overview of Tags

Oracle Cloud Infrastructure supports two kinds of tags: free-form tags and defined tags.

> **Tip**
>
> Watch a video to introduce you to the concepts and features of tagging: Introduction to Tagging.

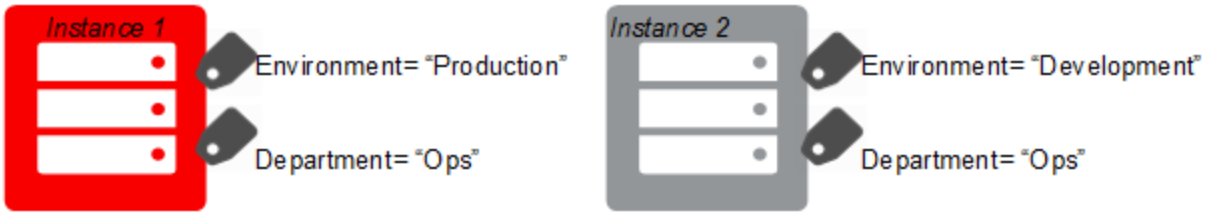
Free-form Tags

Free-form tags consist simply of a key and a value, for example:

Environment: Production

where "Environment" is the key and "Production" is the value.

You can apply multiple free-form tags to a single resource (up to the limit).



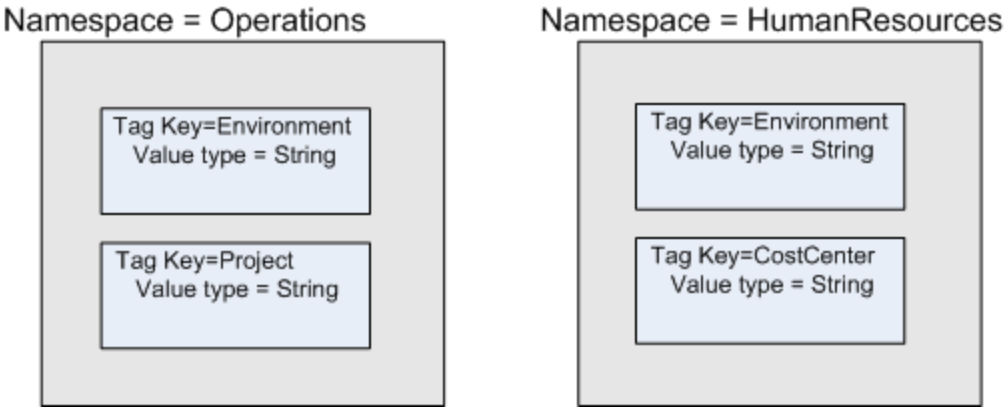
Because free-from tags are limited in functionality, Oracle recommends you only use them when you are first getting started with tagging, to try out the tagging feature in your system. For more information about the features and limitations of free-form tags, see Working with Free-form Tags.

Defined Tags

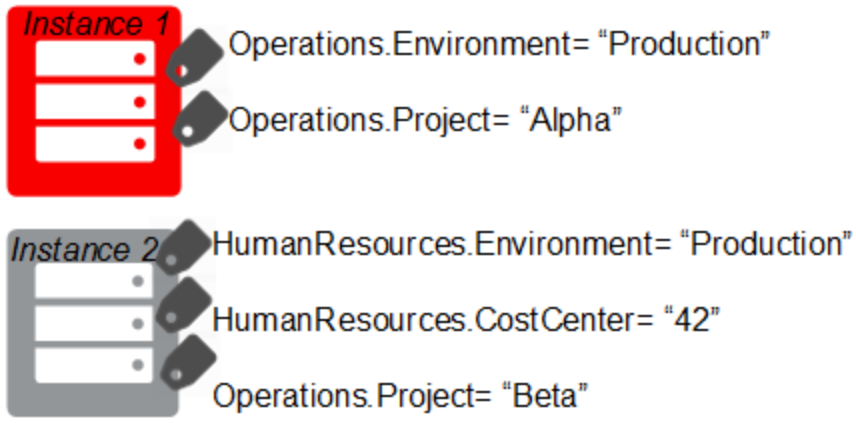
Defined tags provide more features and control than free-form tags. Before you create a defined tag key, you first set up a tag *namespace* for it. You can think of the tag namespace as a container for a set of tag keys. Defined tags support policy to allow you to control who can apply your defined tags. The tag namespace is the entity to which you can apply policy.

To apply a defined tag to a resource, a user first selects the tag namespace, then the tag key within the namespace, and then they can assign the value. Administrators can control which groups of users are allowed to use each namespace.

The following diagrams illustrate defined tags. Two tag namespaces are set up: Operations and HumanResources. The tag keys are defined in the namespaces. Within each namespace, the tag keys must be unique, but a tag key name can be repeated across namespaces. In the example, both namespaces include a key named "Environment".



The first instance is tagged with two tags from the Operations tag namespace, indicating that it belongs to the Operations production environment and the Operations project "Alpha". The second instance is tagged with tags from both the HumanResources tag namespace and the Operations tag namespace, indicating that it belongs to the HumanResources "Production" environment, the HumanResources cost center "42", and also the Operations project "Beta".



Tagging Concepts

Here's a list of the basic tagging concepts:

TAG NAMESPACE

You can think of a tag *namespace* as a container for your tag keys. It consists of a name, and zero or more tag key definitions. Tag namespaces are not case sensitive, and must be unique across the tenancy. The namespace is also a natural grouping to which administrators can apply policy. One policy on the tag namespace applies to all the tag definitions contained in it.

KEY

The name you use to refer to the tag. Tag keys are case insensitive (for example," mytagkey" duplicates "MyTagKey"). Tag keys for defined tags must be created in a namespace. A tag key must be unique within a namespace.

TAG VALUE TYPE

The tag value type specifies the data type allowed for the value. Currently the only data type supported is string.

KEY DEFINITION

A key definition defines the schema of a tag and includes a namespace, tag key, and tag value type.

TAG VALUE

The tag value is the value you give to the tag key. In the example:

Operations.CostCenter="42"

Operations is the namespace, CostCenter is the tag key, and 42 is the tag value. A tag value is optional. Tag values only support the string data type.

TAG (OR DEFINED TAG)

A tag is the instance of a key definition that is applied to a resource. It consists of a namespace, a key and a value. "Tag" is used generically to refer to defined tags.

FREE-FORM TAG

A basic metadata association that consists of a key and a value only. Free-form tags have limited functionality. See Working with Free-form Tags.

COST-TRACKING

Cost tracking is a feature supported for defined tags. Tags enabled for cost tracking (also called cost-tracking tags) allow you use a tag to filter and subtotal your online statement to track usage and costs. For example, you could tag a group of resources with the "Finance.CostCenter" tag and then see the cost generated by just those resources. You view your online statement in My Services.

TAG DEFAULT

Tag defaults let you specify tags to be applied automatically to all resources, in a specific compartment, at the time of creation, regardless of the permissions of the user who creates the resource. See Managing Tag Defaults.

RETIRE

You can't delete a tag key definition or a tag namespace. Instead, you *retire* them. Retired tag namespaces and key definitions can no longer be applied to resources. However, retired

tags are not removed from the resources to which they have already been applied. You can still specify retired tags when searching, filtering, reporting, and so on.

REACTIVATE

You can *reactivate* a tag namespace or tag key definition that has been retired to reinstate its usage in your tenancy.

Taggable Resources

The following table lists resources that support tagging. This table will be updated as tagging support is added for more resources.

Service	Taggable Resource Types
Block Volume	volumes volume_backups volume_groups volume_group_backups
Compute	instance instance-image auto-scaling-configurations instanceconsoleconnections
Database	db-systems databases
Email Delivery	approved-sender

Service	Taggable Resource Types
File Storage	filesystem mount-targets snapshots
Health Checks	health-check-monitor
IAM	groups dynamic-groups users compartments tenancy (root compartment) policies tag-namespaces tag-definitions (API only) identity-providers
Key Management	keys vaults
Load Balancing	load-balancers

Service	Taggable Resource Types
Networking	vcns route-tables security-lists dhcp-options subnets vnics private-ips public-ips internet-gateways local-peering-gateways nat-gateways service-gateways drgs cpes ipsec-connections
Object Storage and Archive Storage	buckets
Resource Manager	jobs stacks
Traffic Management Steering Policies	dns-steering-policies
WAF	waas-policy waas-certificate

Working with Free-form Tags

Free-form tags consist simply of a key-value pair. Free-form tags have limited features. To experience the full feature set of tagging, use defined tags.

Features of free-form tags include:

- Consist of a key and a value. Free-form tags do not belong to a namespace.
- You can apply free-form tags during resource creation or to an existing resource.
- Free-form tag keys are case sensitive. For example, "Project" and "project" are distinct keys.
- Free-form tag values are case sensitive. For example, "alpha" and "Alpha" are distinct values.

Limitations of free-form tags include:

- When applying a free-form tag, you can't see a list of existing free-form tags, so you don't know what tags and values have already been used.
- You can't see a list of existing free-form tags in your tenancy.
- You can't use free-form tags to control access to resources (that is, you can't include free-form tags in IAM policies).

Required Permissions for Working with Free-form Tags

To apply, update, or remove free-form tags for a resource, you must have the update permission on the resource. For many resources, the update permission is granted with the `use` verb. For example, users who can use instances in CompartmentA, can also apply, update, or remove free-form tags for instances in CompartmentA.

Some resources don't include the update permission with the `use` verb. To allow a group to apply, update, or remove free-form tags for these resources without granting the full permissions of `manage`, you can add a policy statement to grant only the *<RESOURCE>*_ UPDATE permission from the `manage` verb. For example, to allow a group NetworkUsers to work with free-from tags with VCNs in CompartmentA, you could write a policy like:

```
Allow group NetworkUsers to use vcns in compartment CompartmentA
```

```
Allow group NetworkUsers to manage vcns in compartment CompartmentA where request.permission='VCN_
UDPATE'
```

The `inspect` verb for a resource grants permissions to view free-form tags for that resource. So users who can `inspect` instances in CompartmentA can also view any free-form tags applied to the instance.

For information about resource permissions, see Policy Reference.

Working with Defined Tags

You must set up the tag namespace and tag keys in your tenancy before users can apply a defined tag to a resource. As part of the set up process, you must also grant permissions to the user groups that will need to use the namespace.

Features of defined tags include:

- Consist of a tag namespace, a key, and a value.
- The tag namespace and tag key definition must be set up in your tenancy before you can apply a defined tag to a resource.
- When applying a defined tag, you select from the list of defined tag keys.
- You can apply a defined tag during resource creation or to an existing resource.
- Defined tag keys are case insensitive.
- Defined tag values are case sensitive. For example, "alpha" and "Alpha" are distinct values.

Required Permissions for Working with Defined Tags

To apply, update, or remove defined tags for a resource, a user must be granted permissions on the resource *and* permissions to use the tag namespace.

Users must be granted `use` access on the *tag namespace* to apply, update, or remove a defined tag for a resource.

Some example policies for tag namespaces:

To allow a group to simply view the tag namespaces in the tenancy (or in a compartment) requires `inspect` access:

```
Allow group GroupA to inspect tag-namespaces in tenancy
```



Important

To apply tags to a resource when using the Console, a user must have permissions to `inspect tag-namespaces in tenancy`. If the user does not have this permission, the list of tag namespaces cannot be presented to the user in the dialog menu.

To allow a group to read the tag definitions contained in tag namespaces requires `read` access:

```
Allow group GroupA to read tag-namespaces in tenancy
```

To allow a group to apply, update, or remove a defined tag for a resource requires the `use` access on the tag namespace:

```
Allow group GroupA to use tag-namespaces in tenancy
```

To allow usage of a specific tag namespace or namespaces , use a `where` clause with the `target.tag-namespace.name` variable. For example:

```
Allow group GroupA to use tag-namespaces in tenancy where target.tag-namespace.name='Operations'
```

or to specify multiple tag namespaces:

```
Allow group GroupA to use tag-namespaces in tenancy where any {target.tag-namespace.name='Operations',
target.tag-namespace.name='HumanResources'}
```

To manage tag namespaces and the tag definitions in them, requires `manage` access:

```
Allow group GroupA to manage tag-namespaces in tenancy
```

In addition to the permissions to work with the tag namespace, to apply or remove defined tags on a resource you must have the update permission for the resource. For many resources, the update permission is granted with the `use` verb. For example, users who can

use instances in CompartmentA, can also apply, update, or remove defined tags for instances in CompartmentA.

Some resources don't include the update permission with the `use` verb. To allow a group to apply, update, or remove defined tags for these resources without granting the full permissions of `manage`, you can add a policy statement to grant only the *<RESOURCE>*_UPDATE permission from the `manage` verb. For example, to allow a group NetworkUsers to work with defined tags with VCNs in CompartmentA, you could write a policy like:

```
Allow group NetworkUsers to use vcns in compartment CompartmentA

Allow group NetworkUsers to manage vcns in compartment CompartmentA where request.permission='VCN_
UDPATE'
```

The `inspect` permission for a resource grants permissions to view defined tags for that resource. For example, users who can `inspect` instances can also view any defined tags applied to the instance.

For information about resource permissions, see Policy Reference.

Example Scenario

Your company has an Operations department. Within the Operations department are several cost centers. You want to be able to tag resources that belong to the Operations department with the appropriate cost center.

1. Create a tag namespace definition called Operations.
2. In the Operations namespace, create a tag key definition called CostCenter.

Alice already belongs to the group InstanceLaunchers. Alice can manage instances in CompartmentA. You want Alice and other members of the InstanceLaunchers group to be able to apply the Operations.CostCenter tag to instances in CompartmentA.

To grant the InstanceLaunchers group access to the Operations tag namespace (and only the Operations tag namespace), add the following statements to the InstanceLaunchers policy:

```
Allow group InstanceLaunchers to use tag-namespaces in compartment CompartmentA where target.tag-
namespace.name='Operations'
```

Alice can now apply the Operations.CostCenter tag to resources in CompartmentA.

Using Cost-Tracking Tags

Cost-tracking tags are displayed in your online billing statement and allow you to filter and subtotal your costs for those resources tagged with a cost-tracking tag. Cost tracking is a feature you can enable for defined tags.

For example, suppose you have a defined tag key definition called CostCenter.Finance. You enable this tag definition for cost tracking. You apply the tag with a value of "W1" (CostCenter.Finance="W1") to some resources, and you apply the tag with a value of "C2" (CostCenter.Finance="C2") to some other resources. When you view your Account Balance and Usage Summary in My Services, you can filter the usage information to show you only the costs generated by the resources tagged with "CostCenter.Finance=W1" . You can then filter your usage summary by "CostCenter.Finance=C2". Using cost-tracking tags, you can compare and track resource usage based on tags.

Enabling Tags for Cost-Tracking

You can enable cost-tracking when you create a tag key definition, or you can update an existing tag key definition to enable cost tracking.

Viewing Usage by Cost-Tracking Tags



Important

After you apply a cost-tracking tag to a resource, allow 5 hours for the usage data to be available on your tenancy's Account Balance and Usage Summary.

Usage tracking by tag begins in the next metering cycle after the tag was applied (or enabled as a cost-tracking tag, if the tag already existed).

To view usage by cost-tracking tags:

1. Navigate to My Services: Open the navigation menu. Under **Solutions, Platform and Edge**, click **My Services Dashboard**.
2. On the My Services dashboard, **Account Management**.
3. On the Usage page, click the Filter by tags text box to select one or more tags. Usage summary is filtered based on all the tags you select.

For more information about this page, see Managing and Monitoring Oracle Cloud.

Limits on Cost-Tracking Tags

- You can have a maximum of 10 tags enabled for cost-tracking in your tenancy at a time. This includes both active and retired tags.
- There may be a delay of up to 5 hours between when a cost-tracking tag is applied to a resource and when the tag information is available in My Services.

Billable Resources That Support Cost-Tracking Tags

You can apply a tag that you have enabled for cost-tracking to any resource that supports tagging. However, not all taggable resources provide cost information to My Services that can be filtered by tag. Currently, the following resources support cost-tracking by tag:

- Compute instances
- Block Volumes
- Object Storage buckets
- Database instances, including Autonomous Data Warehouse and Autonomous Transaction Processing instances

Check back for support for more resources.

For resources that do not yet support cost-tracking by tag, you can still tag the resource with a cost-tracking tag, however, the usage for that resource is not included with the calculations for the tag.

Retiring Key Definitions and Namespace Definitions

You can't delete a tag key definition or a tag namespace definition. Instead, you can retire them.

When you retire a tag key definition, you can no longer apply it to resources. However, the tag is not removed from the resources that it was applied to. The tag still exists as metadata on those resources and you can still call the retired tag in operations (such as listing, sorting, or reporting).



Important

Retiring a tag stops cost tracking for the tag, but if you do not disable the cost-tracking option on the tag key definition, the retired tag continues to count against your maximum of 10 cost-tracking tags for your tenancy. Ensure that you disable cost tracking before you retire the tag key definition. To disable cost-tracking after a tag is retired, you must reactivate the tag key definition to update it. You can't update tag key definitions that are in the retired state.

When you retire a tag namespace, all the tag keys in the tag namespace are retired. As described above, this means that all tags in the tag namespace can no longer be applied to resources, though existing tags are not removed. No new keys can be created in the retired tag namespace.

Reactivating Tag Key Definitions and Namespace Definitions

You can reactivate retired tag key definitions and tag namespace definitions.

When you reactivate a tag key, it is again available for you to apply to resources.

When you reactivate a tag namespace, you can once again create tag key definitions in the namespace. However, if you want to use any of the tag key definitions that were retired with the namespace, you must explicitly reactivate each tag key definition.

Moving Tag Namespaces to a Different Compartment

You can move a tag namespace to a different compartment. The tag namespace can be active or retired when you move it. When you move the tag namespace, all its tag key definitions are moved along with it.

This functionality is useful if you need to re-organize your compartment hierarchy, or if you need to delete a compartment that contains a retired tag namespace. Remember that you can't delete a compartment that still contains resources. A retired tag namespace, even though it is retired, is still an existing resource. Moving the retired tag namespace to a different compartment can enable you to delete its original containing compartment.

To move a tag namespace, you must be allowed to `manage tag-namespaces` in both compartments.

See the procedure To move a tag namespace to a different compartment.

Limits on Tags

See Service Limits for a list of applicable limits and instructions for requesting a limit increase.

Tags per resource: 10 free-form tags and 64 defined tags

Tags enabled for cost-tracking: 10 per tenancy (includes both active and retired tags)

Total tag data size: 5K (JSON). The total tag data size includes all tag data for a single resource (all applied tags and tag values). Sizing is per UTF-8.

Resource	Supported Characters	Max Length
Tag namespace	Printable ASCII, excluding periods (.) and spaces	100 characters
Tag key name (free-form and defined)	Printable ASCII, excluding periods (.) and spaces	100 characters
Tag value (free-form and defined)	Unicode characters	256 characters

Using the Console



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Managing Tag Namespaces

To create a tag namespace

1. Open the navigation menu. Under **Governance and Administration**, go to **Governance** and click **Tag Namespaces**.
   A list of the tag namespaces in your current compartment is displayed.
2. Click **Create Namespace Definition**.

3. Enter the following:
    - **Create in Compartment:** The compartment in which you want to create the namespace definition.
    - **Namespace Definition Name:** A unique name for this set of tags. The name must be unique within your tenancy. Tag namespace is case insensitive. You cannot change this later.
    - **Description:** A friendly description. You can change this later if you want to.
4. Click **Create Namespace Definition**.

To retire a tag namespace

1. Open the navigation menu. Under **Governance and Administration**, go to **Governance** and click **Tag Namespaces**.
   A list of the tag namespaces in your current compartment is displayed.
2. Click the tag namespace you want to retire.
   The namespace's details are displayed.
3. Click **Retire Namespace**.
4. Confirm when prompted.

To reactivate a tag namespace

1. Open the navigation menu. Under **Governance and Administration**, go to **Governance** and click **Tag Namespaces**.
   A list of the tag namespaces in your current compartment is displayed.
2. Click the tag namespace you want to reactivate.
   The namespace's details are displayed.
3. Click **Reactivate Namespace**.
4. Confirm when prompted.

To move a tag namespace to a different compartment

1. Open the navigation menu. Under **Governance and Administration**, go to **Governance** and click **Tag Namespaces**.
   A list of the tag namespaces in your current compartment is displayed.
2. Click the tag namespace you want to move.
   The namespace's details are displayed.
3. Click **Move Tag Namespace**.
4. Select the **Target Compartment** that you want to move the tag namespace to.
5. Click **Move Tag Namespace**.

To delete a tag namespace

You can't delete a tag namespace. To prevent its further use and the further use of all the tag keys that belong to it, you can retire it. If you need to delete the compartment that the tag namespace is in, you can move the tag namespace to a different compartment to enable deletion of the compartment.

Managing Key Definitions

To create a key definition

1. Open the navigation menu. Under **Governance and Administration**, go to **Governance** and click **Tag Namespaces**.
   A list of the tag namespaces in your current compartment is displayed.
2. Click the tag namespace you want to add the tag key definition to.
   A list of the tag key definitions that belong to the namespace is displayed.
3. Click **Create Tag Key Definition**.

4. Enter the following:
    - **Tag Key:** Enter the key. The key can be up to 100 characters in length. Tag keys are case insensitive and must be unique within the tag namespace.
    - **Description:** Enter a friendly description.
    - **Cost-tracking:** Select the check box to enable this tag for cost tracking. You have a limit of 10 cost-tracking tags in your tenancy. For more information, see Using Cost-Tracking Tags.
5. Click **Create Tag Key Definition**.

To update a tag key definition's description

1. Open the navigation menu. Under **Governance and Administration**, go to **Governance** and click **Tag Namespaces**.
   A list of the tag namespaces in your current compartment is displayed.
2. Click the tag namespace that includes the tag key definition you want to update.
   A list of the tag key definitions is displayed.
3. Click the tag key definition you want to update.
   The key definition's details are displayed. The description is displayed under the key definition's name.
4. Click the pencil next to the description.
5. Edit the description and save it.

To enable or disable a tag key definition for cost-tracking

A tag key definition must be active to update it. If you need to disable cost-tracking for a retired tag key definition, you must reactivate it first.

1. Open the navigation menu. Under **Governance and Administration**, go to **Governance** and click **Tag Namespaces**.

A list of the tag namespaces in your current compartment is displayed.

2. Click the tag namespace that includes the tag key definition you want to update.
   A list of the tag key definitions is displayed.
3. Click the tag key definition you want to update.
   The key definition's details are displayed.
4. Click the pencil next to **Cost-tracking**.
5. Select or clear the check box and click **Save**.

To retire a tag key definition

1. Open the navigation menu. Under **Governance and Administration**, go to **Governance** and click **Tag Namespaces**.
   A list of the tag namespaces in your current compartment is displayed.
2. Click the tag namespace that includes the tag key definition you want to retire.
   A list of the tag key definitions is displayed.
3. Click the tag key definition you want to retire.
   The tag key definition's details are displayed. If this is a cost-tracking tag, disable the cost-tracking flag. If you don't disable cost-tracking, this tag will still count against your tenancy maximum of 10 cost-tracking tags, even after it is retired.
4. Click **Retire Tag Key Definition**.
5. Confirm when prompted.

To reactivate a tag key definition

1. Open the navigation menu. Under **Governance and Administration**, go to **Governance** and click **Tag Namespaces**.
   A list of the tag namespaces in your current compartment is displayed.
2. Click the tag namespace that includes the tag key definition you want to reactivate.

A list of the tag key definitions is displayed.

3. Click the tag key definition you want to reactivate.
   The tag key definition's details are displayed.
4. Click **Reactivate Tag Key Definition**.
5. Confirm when prompted.

To delete a tag key definition

You can't delete a tag key definition. To prevent its further use, you can retire it.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage tag namespaces:

- GetTagNamespace
- ListTagNamespaces
- CreateTagNamespace
- UpdateTagNamespace - use to retire or reactivate a tag namespace
- ChangeTagNamespaceCompartment

Use these API operations to manage tag key definitions:

- GetTag - gets the tag key definition
- ListTags - lists tag key definitions
- ListCostTrackingTags - lists the tags that have been enabled for cost-tracking (can be performed in the root compartment only)
- CreateTag - creates a tag key definition

- UpdateTag - updates the tag key definition (use this operation to retire or reactivate a tag key)

Managing Tag Defaults

This topic describes how to you can specify tags to be automatically applied to resources at creation time.

Required IAM Policy

If you're in the Administrators group, then you have the required access for managing tag defaults and tag namespaces. For specific policy information for this feature, see Required Permissions for Working with Tag Defaults.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for tagging or other IAM components, see Details for IAM.

Overview of Tag Defaults

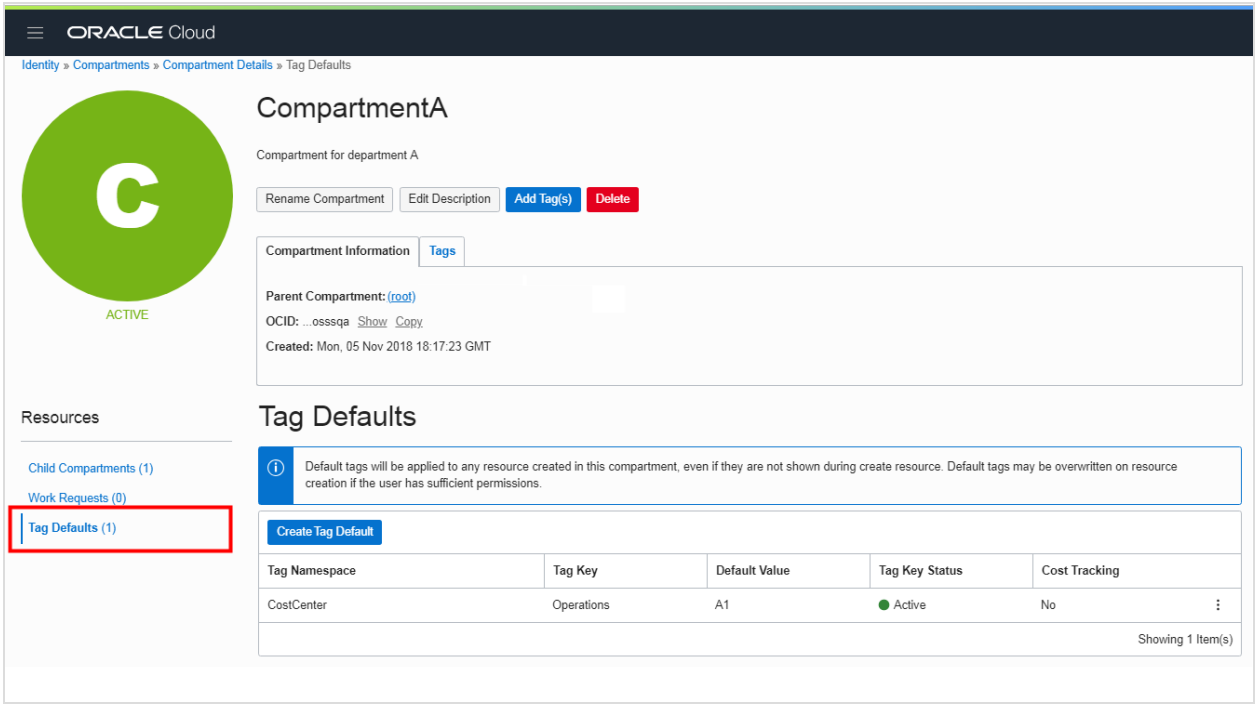
Tag defaults let you specify tags to be applied automatically to all resources, at the time of creation, in a specific compartment. This feature allows you to ensure that appropriate tags are applied at resource creation without requiring the user who is creating the resource to have access to the tag namespaces. Consider the following example:

Alice is a finance administrator and has access to the restricted tag namespace Finance. Alice can set up a tag default to apply the Finance.CostCenter tag to all resources with a value of W1234. Eli can create resources but does not have access to the Finance tag namespace. When Eli creates a resource, the Finance.CostCenter tag is automatically applied with a value of W1234. Eli cannot change this tag, and Alice is confident that it will always be applied correctly and not changed by the users who create or edit resources.

Tag defaults allow tenancy administrators to create secure permissions boundaries between users concerned with governance and users who need to create and administer resources.

Where to Manage Tag Defaults

Tag defaults are defined for a specific compartment, and in the Console you manage them on the Compartment Details page.



Required Permissions for Working with Tag Defaults

To create or edit a tag default for a compartment, you must be granted, at a minimum, the following combination of permissions:

- `manage tag-defaults` access on the compartment where you are adding the tag default
- `use tag-namespaces` access on the compartment where the tag namespace resides
- `inspect tag-namespaces` access on the tenancy

For the full mapping of permissions to API operations, see Details for IAM.

For example, assume you have created a set of tag namespaces in CompartmentA. To give a group named TagAdmins access to add tag defaults to CompartmentA, write a policy with the following statements:

```
Allow group TagAdmins to manage tag-defaults in compartment CompartmentA

Allow group TagAdmins to use tag-namespaces in compartment CompartmentA

Allow group TagAdmins to inspect tag-namespaces in tenancy
```

Now assume you want to allow TagAdmins to also create tag defaults in CompartmentA using tag namespaces that reside in CompartmentZ. Add a statement to allow TagAdmins to use tag namespaces in CompartmentZ:

```
Allow group TagAdmins to use tag-namespaces in compartment CompartmentZ
```

Now when TagAdmins create tag defaults in CompartmentC, they can use tag namespaces that reside in either CompartmentA or CompartmentZ.

Working with Tag Defaults

Tag defaults can only be set up for defined tags. Free-form tags are not supported for tag defaults.

You can define up to 5 tag defaults per compartment.

After a tag default is created, the default tag is applied to any new resources created in the compartment. Previously existing resources in the compartment are not tagged retro-actively. Similarly, if you change the default value of the tag default, existing occurrences are not updated. And, if you delete the tag default from the compartment, existing occurrences of the tag are not removed from resources.

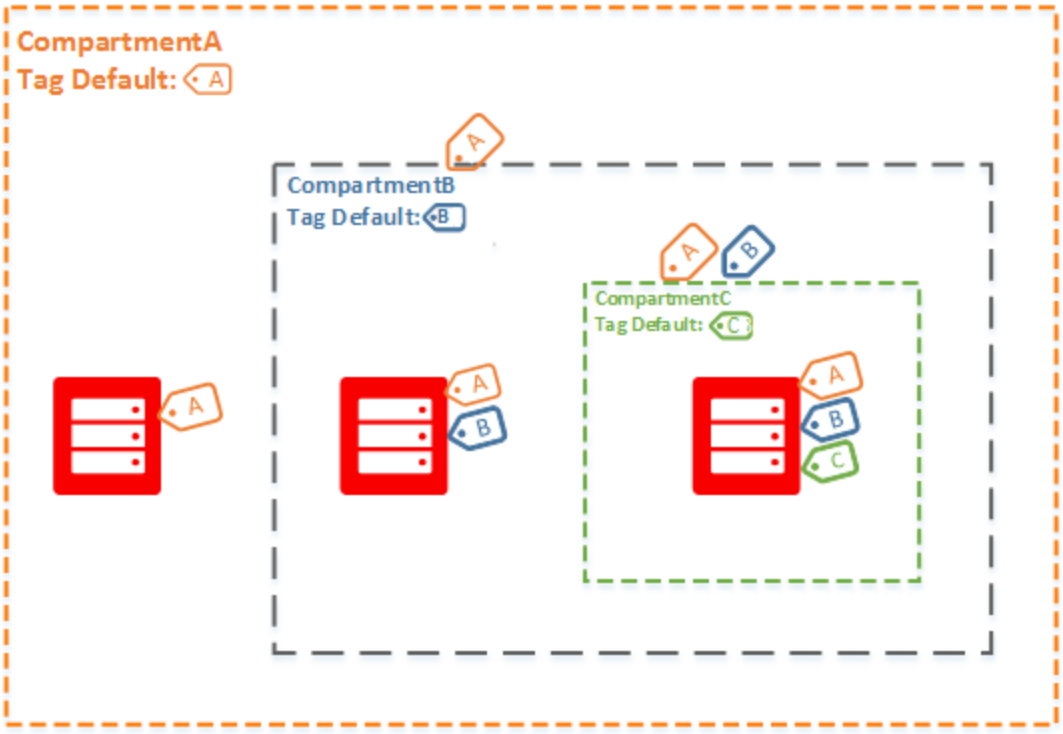
TAG INHERITANCE

The default tag is applied to all resources that get created in the compartment, including child compartments and the resources created in the child compartments.

Example:

- In CompartmentA, you create a tag default, TagA.
  Resources (and compartments) created in CompartmentA are automatically tagged with TagA.
    - In the subcompartment, CompartmentB, you create tag default, TagB.
      Resources and compartments created in CompartmentB are automatically tagged with TagA and TagB.
        - In the sub-subcompartment, CompartmentC, you create tag default TagC.
          Resources created in CompartmentC are automatically tagged with TagA, TagB, and TagC.

This example is illustrated in the following graphic:



Overriding Tag Defaults

Tag defaults can be overridden at the time of resource creation by users who have the appropriate permissions to both create the resource and use the tag namespace.

Example: CompartmentA has a tag default defined to apply CostCenter.Operations="42". Pradeep belongs to a group that grants him permissions to create instances in CompartmentA and also to use tag namespaces in CompartmentA. He creates an instance in CompartmentA, and in the Create Instance dialog, he applies the tag CostCenter.Operations="50". Because he has the appropriate permissions, when the instance is created, the tag default is overridden, and the instance is tagged with CostCenter.Operations="50".

After a resource is created and tagged, users with the appropriate permissions to both update the resource and use the tag namespace can modify the default tags that were applied at resource creation.

Tag Defaults and Retired Tags

Retired tags can't be applied to new resources. Therefore, if the tag namespace or tag key specified in a tag default is retired, resources can no longer be created in the compartment, because the retired tag cannot be applied. You must delete the tag default that specifies the retired tag to continue to create resources in the compartment.

Limits on Tag Defaults

There is a limit of 5 tag defaults that can be defined per compartment.

See Limits on Tags for more limits on tags.

Using the Console to Manage Tag Defaults



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

To create a tag default

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Compartments**.
   A list of the compartments you have access to is displayed.
2. In the list, find the name of the compartment you want to add a tag default to and click its name.
3. On the compartment details page, click **Tag Defaults**.
   The list of existing tag defaults is displayed.
4. Click **Create Tag Default**.
5. Enter the following (all fields are required):
   - **Tag Namespace:** Select the tag namespace for the tag default.
   - **Tag Key:** Select the tag key.
   - **Default Value:** Enter the value you want this tag to have.
6. Click **Create Tag Default**.

To update the default value of a tag default

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Compartments**.

A list of the compartments you have access to is displayed.

2. In the list, click the name of the compartment that has the tag default you want to update.
3. On the compartment details page, click **Tag Defaults**.
   The list of existing tag defaults is displayed.
4. Find the tag default you want to update. Go to the the Actions icon (three dots) and click **Edit**.
5. Enter the new value for the **Default Value**.
6. Click **Save Changes**.

To delete a tag default

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Compartments**.
   A list of the compartments you have access to is displayed.
2. In the list, click the name of the compartment that has the tag default you want to delete.
3. On the compartment details page, click **Tag Defaults**.
   The list of existing tag defaults is displayed.
4. Find the tag default you want to delete. Go to the the Actions icon (three dots) and click **Delete**.
5. Confirm when prompted.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage tag defaults:

- GetTagDefault
- ListTagDefaults
- CreateTagDefault
- UpdateTagDefault
- DeleteTagDefault

Managing Regions

This topic describes the basics of managing your region subscriptions. For more information about regions in Oracle Cloud Infrastructure, see Regions and Availability Domains.

Required IAM Policy

If you're in the Administrators group, then you have the required access to manage region subscriptions.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for managing regions or other IAM components, see Details for IAM.

The Home Region

When you sign up for Oracle Cloud Infrastructure, Oracle creates a tenancy for you in one region. This is your *home region*. Your home region is where your IAM resources are defined. When you subscribe to another region, your IAM resources are available in the new region, however, the master definitions reside in your home region and can only be changed there.

Resources that you can create and update only in the home region are:

- Users
- Groups
- Policies
- Compartments

- Dynamic groups
- Federation resources
- Tag namespaces
- Tag keys

When you use the API to update your IAM resources, you must use the endpoint for your home region. IAM automatically propagates the updates to all regions in your tenancy.

When you use the Console to update your IAM resources, the Console sends the requests to the home region for you. You don't need to switch to your home region first. IAM then automatically propagates the updates to all regions in your tenancy.

When you subscribe your tenancy to a new region, all the policies from your home region are enforced in the new region. If you want to limit access for groups of users to specific regions, you can write policies to grant access to specific regions only. For an example policy, see Restrict admin access to a specific region.



Note

IAM Updates Are Not Immediate Across All Regions

When you create or update an IAM resource, be aware that you need to allow up to several minutes for the changes in your home region to become available in all regions.

Using the Console

To view the list of regions

Open the Console, open the **Region** menu, and then click **Manage Regions**.
A list of the regions offered by Oracle Cloud Infrastructure is displayed. Regions that you

have not subscribed to provide a button to create a subscription.

To subscribe to a region

1. Open the Console, open the **Region** menu, and then click **Manage Regions**.
   The list of regions available to your tenancy Oracle Cloud Infrastructure is displayed. Your home region is labeled.
2. Locate the region you want to subscribe to and click **Subscribe to Region**.
   Note that it could take several minutes to activate your tenancy in the new region.
   Remember, your IAM resources are global, so when the subscription becomes active, all your existing policies are enforced in the new region.
   To switch to the new region, use the **Region** menu in the Console. See Switching Regions for more information.

You cannot unsubscribe from a region.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage regions:

- GetTenancy
- ListRegions: Returns a list of regions offered by Oracle Cloud Infrastructure.
- CreateRegionSubscription
- ListRegionSubscriptions

You cannot unsubscribe from a region.

Region FAQs

Can an individual user subscribe to a region?

A region subscription is at the tenancy level. An administrator can subscribe the tenancy to a region. All IAM polices are enforced in the new region, so *all* users in the tenancy will have the same access and permissions in the new region.

Can I see my existing resources in the new region?

When you select a region in the Console, you are shown a view of the resources in your selected region. Most cloud resources (instances, VCNs, buckets, etc.) exist only in a specific region, so you only see them when you select the region where they were created. The exception is IAM resources: compartments, users, groups, and policies are global across all regions. See also Working in Multiple Regions.

How do my service limits apply to the new region?

Service limits can be scoped to the tenant level, the region level, or the availability domain level. When you subscribe to a new region, you get access to the region and its three availability domains. Service limits apply accordingly. The service limits page lists the scope of each resource limit.

Can I restrict access to a specific region?

Yes. You can write policies that grant permissions in a specified region only. For an example policy, see Restrict admin access to a specific region.

Can I change my home region?

No. Oracle assigns your home region and you can't change it.

Managing the Tenancy

This topic describes options on the tenancy details page in the Console.

Required IAM Policy

If you're in the Administrators group, then you have the required access to manage the tenancy.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for your tenancy and other IAM components, see Details for IAM.

Viewing the Tenancy Details Page

To view the tenancy details page:

Open the **User** menu () and click **Tenancy:** *<your_tenancy_name>*.

Details About Your Tenancy

The tenancy details page provides the following information about your tenancy:

TENANCY OCID

Every Oracle Cloud Infrastructure resource has an Oracle-assigned unique ID called an Oracle Cloud Identifier (OCID). You need your tenancy's OCID to use the API. You'll also need it when contacting support.

HOME REGION

When you sign up for Oracle Cloud Infrastructure, Oracle creates a tenancy for you in one of the available regions. This is your *home region*. Your home region is where your IAM resources are defined. For more information about the home region, see The Home Region.

Audit Retention Period

The retention period for the Audit service logs. The value of the retention period setting affects all regions and all compartments for this tenancy. You can't set different retention periods for different regions or compartments. For more information about this setting, see Setting Audit Log Retention Period.

Object Storage Designated Compartments and Namespace

The Object Storage service provides API support for both Amazon S3 Compatibility API and Swift API. By default, buckets created using the Amazon S3 Compatibility API or the Swift API are created in the root compartment of the Oracle Cloud Infrastructure tenancy. You can designate a different compartment for the Amazon S3 Compatibility API or Swift API to create buckets in. For more information, see Designating Compartments for the Amazon S3 Compatibility and Swift APIs.

For information about your Object Storage namespace, see Understanding Object Storage Namespaces.

Tags

Tagging allows you to define keys and values and associate them with resources. You can then use the tags to help you organize and list resources based on your business needs. If you have permissions to manage the tenancy, you also have permissions to apply free-form tags. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags.

Region Subscriptions and Available Regions

The regions that your tenancy is currently subscribed to and the regions that are available for subscription. For more information about regions, see Regions and Availability Domains and Managing Regions.

Service Limits

The limits allotted to your tenancy and usage against these limits. Not all service resources are included in the list shown here on the Console. For more information or to request an increase, see Service Limits.

Using the API

Many of the options set on this page are managed through the owning service. For example, the Object Storage settings are managed with the Object Storage service API, and setting the Audit log retention period is handled by the Audit service API.

To get information about your tenancy use the following operation:

- GetTenancy

To tag a tenancy, use the following operations:

- GetCompartment
- UpdateCompartment

In the above operations, use the tenancy OCID for the `compartmentID` parameter.

Managing Policies

This topic describes the basics of working with policies.

Required IAM Policy

If you're in the Administrators group, then you have the required access for managing policies.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies to control who else can write policies or manage other IAM components, see Let a compartment admin manage the compartment, and also Details for IAM.

Tagging Resources

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

Working with Policies

If you haven't already, make sure to read How Policies Work to understand the basics of how policies work.

When creating a policy, you must specify the compartment where it should be *attached*, which is either the tenancy (the root compartment) or another compartment. Where it's attached governs who can later modify or delete it. For more information, see Policy Attachment. When creating the policy in the Console, you attach the policy to the desired compartment by creating the policy *while viewing that compartment*. If you're using the API, you specify the identifier of the desired compartment in the CreatePolicy request.

Also when creating a policy, you can specify its *version date*. For more information, see Policy Language Version. You can change the version date later if you like.

When creating a policy, you must also provide a unique, non-changeable *name* for it. The name must be unique across all policies in your tenancy. You must also provide a *description* (although it can be an empty string), which is a non-unique, changeable description for the policy. Oracle will also assign the policy a unique ID called an Oracle Cloud ID. For more information, see Resource Identifiers.



Note

If you delete a policy and then create a new policy with the same name, they'll be considered different policies because they'll have different OCIDs.

For information about how to write a policy, see How Policies Work and Policy Syntax.

When you create a policy, make changes to an existing policy, or delete a policy, your changes go into effect typically within 10 seconds.

You can view a list of your policies in the Console or with the API. In the Console, the list is automatically filtered to show only the policies attached to the compartment you're viewing. To determine which policies apply to a particular group, you must view the individual statements inside all your policies. There isn't a way to automatically obtain that information in the Console or API.

For information about the number of policies you can have, see Service Limits.

Using the Console

To create a policy

Prerequisite: The group and compartment that you're writing the policy for must already exist.

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Policies**.
   A list of the policies in the compartment you're viewing is displayed.
2. If you want to attach the policy to a compartment other than the one you're viewing, select the desired compartment from the list on the left. Where the policy is attached controls who can later modify or delete it (see Policy Attachment).
3. Click **Create Policy**.
4. Enter the following:
   - **Name:** A unique name for the policy. The name must be unique across all policies in your tenancy. You cannot change this later.
   - **Description:** A friendly description. You can change this later if you want to.
   - **Policy Versioning:** Select **Keep Policy Current** if you'd like the policy to stay current with any future changes to the service's definitions of verbs and resources. Or if you'd prefer to limit access according to the definitions that were current on a specific date, select **Use Version Date** and enter that date in format YYYY-MM-DD format. For more information, see Policy Language Version.
   - **Statement:** A policy statement. For the correct format to use, see Policy Basics and also Policy Syntax. If you want to add more than one statement, click **+**.
   - **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
5. Click **Create**.

The new policy will go into effect typically within 10 seconds.

To get a list of your policies

Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Policies**. A list of the policies in the compartment you're currently viewing is displayed. If you want to view policies attached to a different compartment, select that compartment from the list on the left. You can't get a single list of all policies; they're always displayed by compartment.

To determine which policies apply to a particular group, you must view the individual statements inside all your policies. There isn't a way to automatically obtain that information in the Console.

To update the description for an existing policy

This is available only through the API. A workaround is to create a new policy with the new description and delete the old policy.

To update the statements in an existing policy

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Policies**.
   A list of the policies in the compartment you're viewing is displayed. If you don't see the one you're looking for, verify that you're viewing the correct compartment (select from the list on the left side of the page).
2. Click the policy you want to update.
   The policy's details and statements are displayed.
3. Either delete or add new statements (for the required format for statements, see Policy Basics and Policy Syntax). If you want to update an existing statement, create a new one with your desired changes and then delete the old one.

Your changes will go into effect typically within 10 seconds.

To update the version date for an existing policy

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Policies**.
   A list of the policies in the compartment you're currently viewing is displayed. If you don't see the policy you're looking for, make sure you're viewing the correct compartment (select from the list on the left side of the page).
2. Click the policy you want to update.
   The policy's details, version date, and statements are displayed.
3. Click **Update Version Date**.
4. Select **Keep Policy Current** if you'd like the policy to stay current with any future changes to the service's definitions of verbs and resources. Or if you'd prefer to limit access according to the definitions that were current on a specific date, select **Use Version Date** and enter that date in format YYYY-MM-DD format. For more information, see Policy Language Version.
5. Click **Update Version Date**.

Your changes will go into effect typically within 10 seconds.

To delete a policy



Tip

Remember that if you delete a policy and then create a new one with the same name, they'll be considered different policies because they'll have different OCIDs.

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Policies**.
   A list of the policies in the compartment you're viewing is displayed. If you don’t see

the one you're looking for, verify that you're viewing the correct compartment (select from the list on the left side of the page).

2. For the policy you want to delete, click **Delete**.
3. Confirm when prompted.

Your changes will go into effect typically within 10 seconds.

To apply tags to a policy

For instructions, see Resource Tags.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.



Note

Updates Are Not Immediate Across All Regions

Your IAM resources reside in your home region. To enforce policy across all regions, the IAM service replicates your resources in each region. Whenever you create or change a policy, user, or group, the changes take effect first in the home region, and then are propagated out to your other regions. It can take several minutes for changes to take effect in all regions. For example, assume you have a group with permissions to launch instances in the tenancy. If you add UserA to this group, UserA will be able to launch instances in your home region within a minute. However, UserA will not be able to launch instances in other regions until the replication process is complete. This process can take up to several minutes. If UserA tries to launch an instance before replication is complete, they will get a not authorized error.

Use these API operations to manage policies:

- CreatePolicy
- ListPolicies
- GetPolicy
- UpdatePolicy
- DeletePolicy

Managing User Credentials

This topic describes the basics of working with Oracle Cloud Infrastructure Identity and Access Management (IAM) user credentials. If you're not already familiar with the available credentials, see User Credentials.

Working with Console Passwords and API Keys

Each user automatically has the ability to change or reset *their own* Console password, as well as manage *their own* API keys. An administrator does not need to create a policy to give a user those abilities.

To manage credentials for users other than yourself, you must be in the Administrators group or some other group that has permission to work with the tenancy. Having permission to work with a compartment within the tenancy is not sufficient. For more information, see The Administrators Group and Policy.

IAM administrators (or anyone with permission to the tenancy) can use either the Console or the API to manage all aspects of both types of credentials, for themselves and all other users. This includes creating an initial one-time password for a new user, resetting a password, uploading API keys, and deleting API keys.

Users who are not administrators can manage *their own* credentials. In the Console, users can:

- Change or reset their own password.
- Upload an API key in the Console for their own use (and also delete their own API keys).

And with the API, users can:

- Reset their own password with CreateOrResetUIPassword.
- Upload an additional API key to the IAM service for their own use with UploadApiKey (and also delete their own API keys with DeleteApiKey). Remember that a user can't use the API to change or delete their own credentials until they themselves upload a key in the Console, or an administrator uploads a key for that user in the Console or the API.

A user can have a maximum of three API keys at a time.

Working with Auth Tokens



Note

"Auth tokens" were previously named "Swift passwords". Any Swift passwords you had created are now listed in the Console as auth tokens. You can continue to use the existing passwords.

Auth tokens are Oracle-generated token strings that you can use to authenticate with third-party APIs that do no support Oracle Cloud Infrastructure's signature-based authentication. Each user created in the IAM service automatically has the ability to create, update, and delete their own auth tokens in the Console or the API. An administrator does not need to create a policy to give a user those abilities. Administrators (or anyone with permission to the tenancy) also have the ability to manage auth tokens for other users.

Note that you cannot change your auth token to a string of your own choice. The token is always an Oracle-generated string.

Auth tokens do not expire. Each user can have up to two auth tokens at a time. To get an auth token in the Console, see To create an auth token.

Using an Auth Token with Swift

Swift is the OpenStack object store service. If you already have an existing Swift client, you can use it with the Recovery Manager (RMAN) to back up an Oracle Database System (DB System) database to Object Storage. You will need to get an auth token to use as your Swift password. When you sign in to your Swift client, you provide the following:

- Your Oracle Cloud Infrastructure Console user login
- Your Swift-specific auth token, provided by Oracle

- Your organization's Oracle tenant name

Any user of a Swift client that integrates with Object Storage needs permission to work with the service. If you're not sure if you have permission, contact your administrator. For information about policies, see How Policies Work. For basic policies that enable use of Object Storage, see Common Policies.

Working with Customer Secret Keys



Note

"Customer Secret keys" were previously named "Amazon S3 Compatibility API keys". Any keys you had created are now listed in the Console as Customer Secret keys. You can continue to use the existing keys.

Object Storage provides an API to enable interoperability with Amazon S3. To use this Amazon S3 Compatibility API, you need to generate the signing key required to authenticate with Amazon S3. This special signing key is an Access Key/Secret Key pair. Oracle provides the Access Key that is associated with your Console user login. You or your administrator generates the Customer Secret key to pair with the Access Key.

Each user created in the IAM service automatically has the ability to create, update, and delete their own Customer Secret keys in the Console or the API. An administrator does not need to create a policy to give a user those abilities. Administrators (or anyone with permission to the tenancy) also have the ability to manage Customer Secret keys for other users.

Any user of the Amazon S3 Compatibility API with Object Storage needs permission to work with the service. If you're not sure if you have permission, contact your administrator. For information about policies, see How Policies Work. For basic policies that enable use of Object Storage, see Common Policies.

Customer Secret keys do not expire. Each user can have up to two Customer Secret keys at a time. To create keys using the Console, see To create a Customer Secret key.

Working with SMTP Credentials

Simple Mail Transfer Protocol (SMTP) credentials are needed in order to send email through the Email Delivery service. Each user is limited to a maximum of two SMTP credentials. If more than two are required, they must be generated on other existing users or additional users must be created.



Note

You cannot change your SMTP username or password to a string of your own choice. The credentials are always Oracle-generated strings.

Each user created in the IAM service automatically has the ability to create and delete their own SMTP credentials in the Console or the API. An administrator does not need to create a policy to give a user those abilities. Administrators (or anyone with permission to the tenancy) also have the ability to manage SMTP credentials for other users.



Tip

Although each user can create and delete their own credentials, it is a security best practice to create a new user and generate SMTP credentials on this user rather than generating SMTP credentials on your Console user that already has permissions assigned to it.

SMTP credentials do not expire. Each user can have up to two credentials at a time. To get SMTP credentials in the Console, see To generate SMTP credentials.

For information about using the Email Delivery service, see Overview of the Email Delivery Service.

Using the Console

To change your Console password

You're prompted to change your initial one-time password *the first time* you sign in to the Console. The following procedure is for changing your password again later.

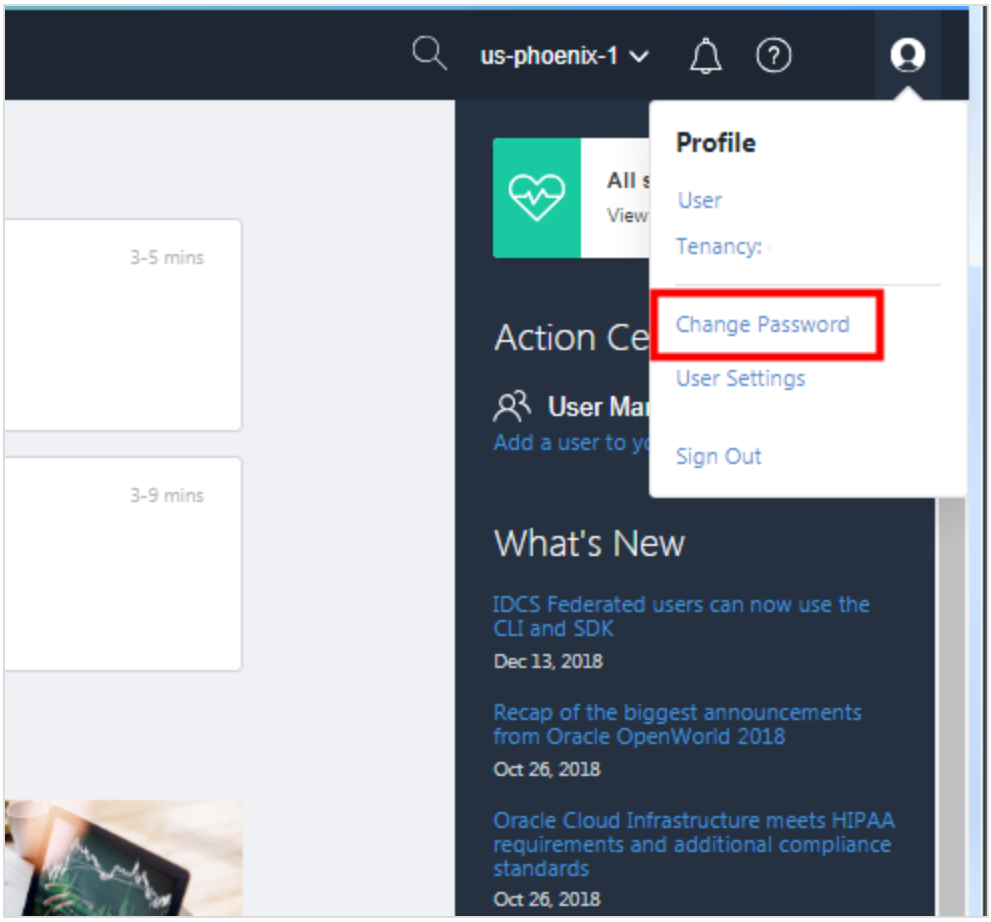


Note

For Federated Users

If your company uses an identity provider (other than Oracle Identity Cloud Service) to manage user logins and passwords, you can't use the Console to update your password. You do that with your identity provider.

1. Sign in to the Console using the Oracle Cloud Infrastructure Username and Password.
2. After you sign in, go to the top-right corner of the Console, open the **User** menu () and then click **Change Password**.



3. Enter the current password.
4. Follow the prompts to enter the new password, and then click **Save New Password**.

To create or reset another user's Console password

If you're an administrator, you can use the following procedure to create or reset a user's password. The procedure generates a new one-time password that the user must change the next time they sign in to the Console.

1. View the user's details: In the Console, click **Identity**, and then click **Users**. Locate the user in the list, and then click the user's name to view the details.
2. Click **Create/Reset Password**.
   The new one-time password is displayed. If you're an administrator performing the task for another user, you need to securely deliver the new password to the user. The user will be prompted to change their password the next time they sign in to the Console. If they don't change it within 7 days, the password will expire and you'll need to create a new one-time password for the user.

To reset your password if you forgot it

If you have an email address in your user profile, you can use the **Forgot Password** link on the sign on page to have a temporary password sent to you. If you don't have an email address in your user profile, you must ask an administrator to reset your password for you.

To unblock a user

If you're an administrator, you can unblock a user who has tried 10 times in a row to sign in to the Console unsuccessfully. See To unblock a user.

To upload an API signing key

The following procedure works for a regular user or an administrator. Administrators can upload an API key for either another user or themselves.



Important

The API key must be an **RSA key in PEM format (minimum 2048 bits)**. The PEM format looks something like this:

```
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAoTFqF...
...
-----END PUBLIC KEY----
```

For more information about generating a public PEM key, see Required Keys and OCIDs.

1. View the user's details:
   - If you're uploading an API key for *yourself*: Open the **User** menu () and click **User Settings**.
   - If you're an administrator uploading an API key for *another user*: In the Console, click **Identity**, and then click **Users**. Locate the user in the list, and then click the user's name to view the details.
2. Click **Add Public Key**.
3. Paste the key's value into the window and click **Add**.
   The key is added and its fingerprint is displayed (example fingerprint: d1:b2:32:53:d3:5f:cf:68:2d:6f:8b:5f:77:8f:07:13).



Note

When making API requests, you'll need the key's fingerprint, along with your tenancy's OCID and user OCID. See Required Keys and OCIDs.

To delete an API signing key

The following procedure works for a regular user or an administrator. Administrators can delete an API key for either another user or themselves.

1. View the user's details:
   - If you're deleting an API key for *yourself*: Open the **User** menu () and click **User Settings**.
   - If you're an administrator deleting an API key for *another user*: In the Console, click **Identity**, and then click **Users**. Locate the user in the list, and then click the user's name to view the details.
2. For the API key you want to delete, click **Delete**.
3. Confirm when prompted.

The API key is no longer valid for sending API requests.

To create an auth token

1. View the user's details:
   - If you're creating an auth token for yourself: Open the **User** menu () and click **User Settings**.
   - If you're an administrator creating an auth token for another user: In the Console, click **Identity**, and then click **Users**. Locate the user in the list, and then click the user's name to view the details.
2. On the left side of the page, click **Auth Tokens**.
3. Click **Generate Token**.
4. Enter a description that indicates what this token is for, for example, "Swift password token".
5. Click **Generate Token**.
   The new token string is displayed.

6. Copy the token string immediately, because you can't retrieve it again after closing the dialog box.

If you're an administrator creating an auth token for another user, you need to securely deliver it to the user by providing it verbally, printing it out, or sending it through a secure email service.

To delete an auth token

The following procedure works for a regular user or an administrator. Administrators can delete an auth token for either another user or themselves.

1. View the user's details:
   - If you're deleting an auth token for *yourself*: Open the **User** menu () and click **User Settings**.
   - If you're an administrator deleting an auth token for *another user*: In the Console, click **Identity**, and then click **Users**. Locate the user in the list, and then click the user's name to view the details.
2. On the left side of the page, click **Auth Tokens**.
3. For the auth token you want to delete, click **Delete**.
4. Confirm when prompted.

The auth token is no longer valid for accessing third-party APIs.

To create a Customer Secret key

1. View the user's details:
   - If you're creating a Customer Secret key for yourself: Open the **User** menu () and click **User Settings**.

- If you're an administrator creating a Customer Secret key for another user: In the Console, click **Identity**, and then click **Users**. Locate the user in the list, and then click the user's name to view the details.

2. On the left side of the page, click **Customer Secret Keys**.
   A Customer Secret key consists of an Access Key/Secret key pair. Oracle automatically generates the Access Key when you or your administrator generates the Secret Key to create the Customer Secret key.
3. Click **Generate Secret Key**.
4. Enter a friendly description for the key and click **Generate Secret Key**.
   The generated **Secret Key** is displayed in the **Generate Secret Key** dialog box. At the same time, Oracle generates the **Access Key** that is paired with the **Secret Key**. The newly generated Customer Secret key is added to the list of **Customer Secret Keys**.
5. Copy the **Secret Key** immediately, because you can't retrieve the **Secret Key** again after closing the dialog box for security reasons.
   If you're an administrator creating a Secret Key for another user, you need to securely deliver it to the user by providing it verbally, printing it out, or sending it through a secure email service.
6. Click **Close**.
7. To show or copy the **Access Key**, click the **Show** or **Copy** action to the left of the **Name** of a particular Customer Secret key.

To delete a Customer Secret key

The following procedure works for a regular user or an administrator. Administrators can delete a Customer Secret key for either another user or themselves.

1. View the user's details:
   - If you're deleting a Customer Secret key for *yourself*: Open the **User** menu () and click **User Settings**.

- If you're an administrator deleting a Customer Secret key for *another user*: In the Console, click **Identity**, and then click **Users**. Locate the user in the list, and then click the user's name to view the details.

2. On the left side of the page, click **Customer Secret Keys**.
3. For the Customer Secret key you want to delete, click **Delete**.
4. Confirm when prompted.

The Customer Secret key is no longer available to use with the Amazon S3 Compatibility API.

To generate SMTP credentials

1. View the user's details:
   - If you're generating SMTP credentials for *yourself*: Open the **User** menu () and click **User Settings**.
   - If you're an administrator generating SMTP credentials for *another user*: In the Console, click **Identity**, and then click **Users**. Locate the user in the list, and then click the user's name to view the details.
2. Click **SMTP Credentials**.
3. Click **Generate SMTP Credentials**.
4. Enter a **Description** of the SMTP Credentials in the dialog box.
5. Click **Generate SMTP Credentials**. A user name and password is displayed.
6. Copy the user name and password for your records and click **Close**. Copy the credentials immediately, because you can't retrieve the password again after closing the dialog box for security reasons.

   If you're an administrator creating the credential set for another user, you need to securely deliver it to the user by providing it verbally, printing it out, or sending it through a secure email service.

To delete SMTP credentials

The following procedure works for a regular user or an administrator. Administrators can delete SMTP credentials for either another user or themselves.

1. View the user's details:
   - If you're deleting SMTP credentials for *yourself*: Open the **User** menu () and click **User Settings**.
   - If you're an administrator deleting SMTP credentials for *another user*: In the Console, click **Identity**, and then click **Users**. Locate the user in the list, and then click the user's name to view the details.
2. On the left side of the page, click **SMTP Credentials**.
3. For the SMTP credentials you want to delete, click **Delete**.
4. Confirm when prompted.

The SMTP credentials are no longer available to use with the Email Delivery service.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use this API operation to manage Console passwords and access:

- CreateOrResetUIPassword: This generates a new one-time Console password for the user. The next time the user signs in to the Console, they'll be prompted to change the password.
- UpdateUserState: Unblocks a user who has tried to sign in 10 times in a row unsuccessfully.

Use these API operations to manage API signing keys:

- ListApiKeys
- UploadApiKey
- DeleteApiKey

Use these API operations to manage auth tokens:

- CreateAuthToken
- UpdateAuthToken: You can only update the auth token's description, not change the token string itself.
- ListAuthTokens
- DeleteAuthToken

Use these API operations to manage Customer Secret keys:

- CreateCustomerSecretKey
- UpdateCustomerSecretKey: You can only update the secret key's description, not change the key itself.
- ListCustomerSecretKeys
- DeleteCustomerSecretKey

Use these API operations to manage SMTP credentials:

- CreateSmtpCredential
- UpdateSmtpCredential: You can only update the description.
- ListSmtpCredentials
- DeleteSmtpCredential

Managing Authentication Settings

This topic describes how to set password policy rules for local IAM users in your tenancy.

Required IAM Policy

If you're in the Administrators group, then you have the required access for managing password policy.

To view authentication policy, you must be granted `inspect` access on the `authentication-policies` resource. For example:

```
Allow group GroupA to inspect authentication-policies in tenancy
```

To modify authentication policy, you must be granted the `AUTHENTICATION_POLICY_UPDATE` permission. This permission is included in the `manage` verb. For example:

```
Allow group GroupA to manage authentication-policies in tenancy
```

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for groups or other IAM components, see Details for IAM.

Working with Password Policy Rules

A password policy that you set in the IAM service is applicable for all local (or non-federated) users.

When a user is created or when a user changes their password, the IAM service validates the password that is provided against the password policy to ensure that it meets the criteria for the policy. When a user logs in for the first time to change the password, or resets the password at any time, the password policy is evaluated and enforced.

When Do Changes to Password Policy Rules Take Effect

Changes to password policy rules take effect immediately so that the next time any user changes their password they must create a password that meets the criteria. Existing passwords will continue to work even if they would be invalid under the new rules. Users are not forced to change existing passwords to meet the new criteria. Passwords are evaluated against the rules only at the time they are created or changed.

About the Password Policy Rules

The following table describes the rules that you can include in your password policy:

Rule	Setting Options	Default IAM Service Setting
Minimum password length	Minimum value is 8 (characters). Maximum value is 100.	12 characters
Special characters	Require passwords to contain at least 1 of the following special characters: !\"#$%&`()*+,-./:;<=>?@[\\]^_` {|}~	Enforced
Lowercase characters	Require passwords to contain at least 1 lowercase alphabetic character a-z.	Enforced
Uppercase characters	Require passwords to contain at least 1 uppercase alphabetic character A-Z.	Enforced
Numeric characters	Require passwords to contain at least 1 number 0-9.	Enforced

Oracle recommends that you enforce all the password rules.

Using the Console

To edit password policy rules

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Authentication Settings**.

The authentication settings for your tenancy are displayed.

2. Click **Edit**.
3. Enter the following to set the password policy:
    - **Minimum Password Length:** Enter a number to define the minimum number of characters that a user's password must contain. Allowed values are 8 through 100.
4. Select the **Password Rules** you want to enforce:
    - **Must contain at least 1 numeric character:** Select the check box to require at least 1 number (0-9) in the password.
    - **Must contain at least 1 special character:** Select the check box to require at least 1 special character. Special characters are: !\"#$%&`()*+,-./:;<=>?@[\\]^_`{|}~
    - **Must contain at least 1 lowercase character:** Select the check box to require at least 1 lowercase alphabetic character (a-z).
    - **Must contain at least 1 uppercase character:** Select the check box to require at least 1 uppercase alphabetic character (A-Z).
5. Click **Save**.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage password rules:

- GetAuthenticationPolicy
- UpdateAuthenticationPolicy

Managing Multi-Factor Authentication

This topic describes how users can manage multi-factor authentication (MFA) in Oracle Cloud Infrastructure.

Required IAM Policy

Only the user can enable multi-factor authentication (MFA) for their own account. Users can also disable MFA for their own accounts. Members of the Administrators group can disable MFA for other users, but they cannot enable MFA for another user.

About Multi-Factor Authentication

Multi-factor authentication is a method of authentication that requires the use of more than one factor to verify a user's identity.

With MFA enabled in the IAM service, when a user signs in to Oracle Cloud Infrastructure, they are prompted for their user name and password, which is the first factor (something that they know). The user is then prompted to provide a second verification code from a registered MFA device, which is the second factor (something that they have). The two factors work together, requiring an extra layer of security to verify the user's identity and complete the sign-in process.

In general, MFA may include any two of the following:

- Something that you know, like a password.
- Something that you have, like a device.
- Something that you are, like your fingerprint.

The IAM service supports two-factor authentication using a password (first factor) and a device that can generate a time-based one-time password (TOTP) (second factor).

General Concepts

Here's a list of the basic concepts you need to be familiar with.

MULTI-FACTOR AUTHENTICATION (MFA)

Multi-factor authentication (MFA) is a method of authentication that requires the use of more than one factor to verify a user's identity. Examples of authentication factors are a password (something you know) and a device (something you have).

AUTHENTICATOR APP

An app you install on your mobile device that can provide software-based secure tokens for identity verification. Examples of authenticator apps are Oracle Mobile Authenticator and Google Authenticator. To enable MFA for the IAM service, you'll need a device with an authenticator app installed. You'll use the app to register your device and then you'll use the same app (on the same device) to generate a time-based one-time passcode every time you sign in.

REGISTERED MOBILE DEVICE

Multi-factor authentication is enabled for a specific user and for a specific device. The procedure to enable MFA for a user includes the registration of the mobile device. This same device must be used to generate the time-based one-time passcode every time the user signs in. If the registered mobile device becomes unavailable, an administrator must disable MFA for the user so that MFA can be re-enabled with a new device.

TIME-BASED ONE-TIME PASSWORD (TOTP)

A TOTP is a password (or passcode) that is generated by an algorithm that computes a one-time password from a shared secret key and the current time, as defined in RFC 6238. The authenticator app on your registered mobile device generates the TOTP that you need to enter every time you sign in to Oracle Cloud Infrastructure.

Supported Authenticator Apps

The following authenticator apps have been tested with the Oracle Cloud Infrastructure IAM service:

- Oracle Mobile Authenticator
- Google Authenticator

You can find these apps in your mobile device's app store. You must install one of these apps on your mobile device before you can enable MFA.

Working with MFA

Keep the following in mind when you enable MFA:

- You must install a supported authenticator app on the mobile device you intend to register for MFA.
- Each user must enable MFA for themselves using a device they will have access to every time they sign in. An administrator *cannot* enable MFA for another user.
- To enable MFA, you use your mobile device's authenticator app to scan a QR code that is generated by the IAM service and displayed in the Console. The QR code shares a secret key with the app to enable the app to generate TOTPs that can be verified by the IAM service.
- A user can register only one device to use for MFA.
- After you add your Oracle Cloud Infrastructure account to your authenticator app, the account name displays in the authenticator app as Oracle *<tenancy_name>* - *<username>*.

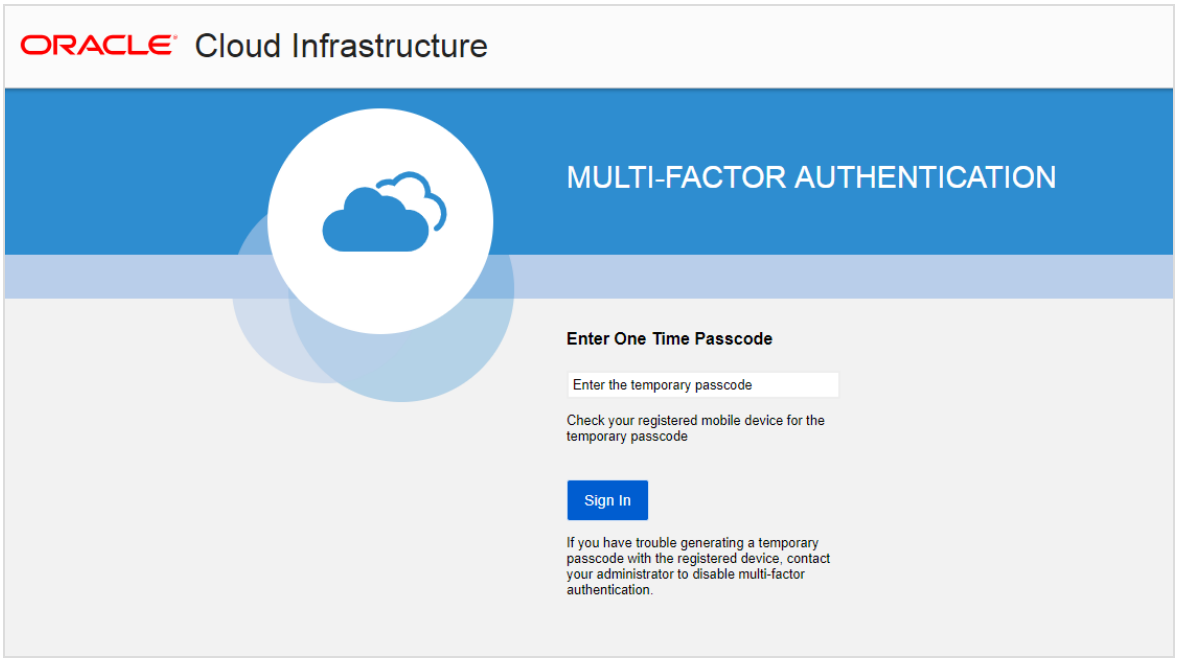
Sign in Process After Enabling MFA

After you have enabled MFA, use one of the following procedures to sign in to Oracle Cloud Infrastructure:

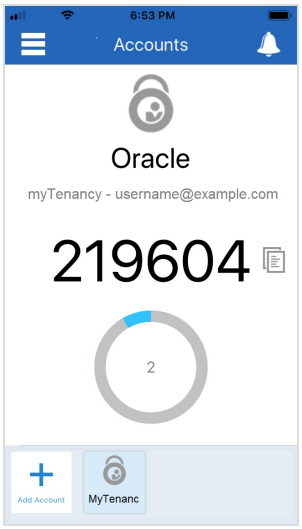
To sign in using the Console

1. Navigate to the Console sign-in page.
2. Enter your Oracle Cloud Infrastructure **User Name** and **Password** and then click **Sign In**.

After your user name and password are authenticated, you have successfully supplied the first factor for authentication. The secondary authentication page displays and prompts you to enter a one-time passcode, as shown in the following screenshot.



3. Open the authenticator app on your registered mobile device and then open the account for your Oracle Cloud Infrastructure tenancy. The following screenshot shows an example from Oracle Mobile Authenticator.



4. Enter the passcode displayed by your authenticator app (for example, 219604) and then click **Sign In**.

   Important: The authenticator app generates a new time-based one-time passcode every 30 seconds. You must enter a code while the code is still valid. If you miss the time window for one passcode, you can enter the next one that is generated. Just ensure that you enter the code that is currently displayed by your app.

To sign in using the command line interface (CLI)

1. To sign in with the CLI, run the following command:

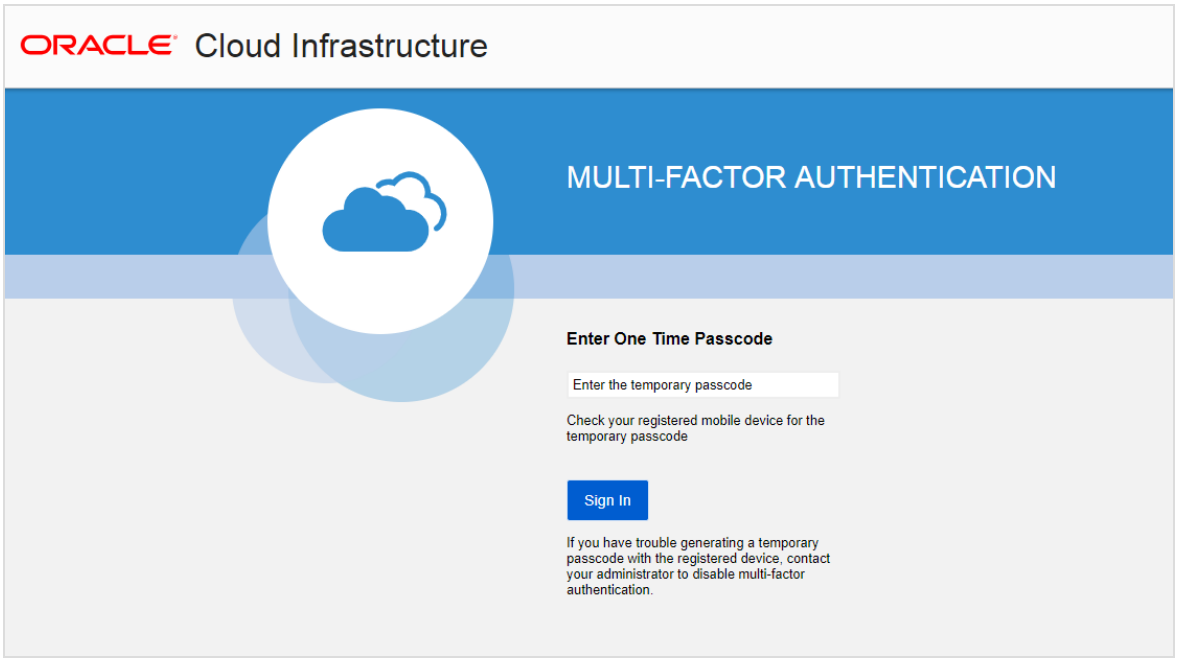
   ```
   oci session authenticate --region us-ashburn-1
   ```

   A browser window opens, and a prompt instructs you to use the browser to sign in.

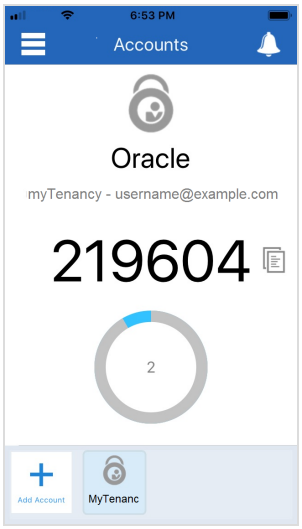
   ```
   Please switch to newly opened browser window to log in!
   ```

2. In the browser window, enter your Oracle Cloud Infrastructure **User Name** and **Password** and then click **Sign In**.

After your user name and password are authenticated, you have successfully supplied the first factor for authentication. The secondary authentication page displays and prompts you to enter a one-time passcode, as shown in the following screenshot.



3. Open the authenticator app on your registered mobile device and then open the account for your Oracle Cloud Infrastructure tenancy. The following screenshot shows an example from Oracle Mobile Authenticator.



4. Enter the passcode displayed by your authenticator app (for example, 219604) and then click **Sign In**.

   Important: The authenticator app generates a new time-based one-time passcode every 30 seconds. You must enter a code while the code is still valid. If you miss the time window for one passcode, you can enter the next one that is generated. Just ensure that you enter the code that is currently displayed by your app.

   After you authenticate, prompts instruct you to return to the CLI and enter the name of a profile.

5. In the CLI, type a name for the profile.

> **Tip**
>
> For more information about working with the CLI, see Quickstart and Getting Started with the Command Line Interface.

What To Do If You Lose Your Registered Mobile Device

If you lose your registered mobile device, you will not be able to authenticate to Oracle Cloud Infrastructure through the Console. Contact your administrator to disable multi-factor authentication for your account. You can then repeat the process to enable multi-factor authentication with a new mobile device.

Unblocking a User After Unsuccessful Sign-in Attempts

If a user tries 10 times in a row to sign in to the Console unsuccessfully, they will be automatically blocked from further sign-in attempts. An administrator can unblock the user in the Console (see To unblock a user) or with the UpdateUserState API operation.

Disabling MFA

Each user can disable MFA for themselves. An administrator can also disable MFA for another user.



Warning

Do not disable MFA unless you are instructed to by your administrator.

Using the Console

Use the following procedures to manage MFA in the Console.

To enable MFA for your user account

Prerequisite: You must install a supported authenticator app on the mobile device you intend to register for MFA.

1. In the upper-right corner of the Console, open the **User** menu () and then select **User Settings**. Your user details are displayed.
2. Click **Enable Multi-Factor Authentication**.
3. Scan the QR code displayed in the dialog with your mobile device's authenticator app.
   Note: If you close the browser, or if the browser crashes before you can enter the verification code, you must generate a new QR code and scan it again with your app. To generate a new QR code, click the **Enable Multi-Factor Authentication** button again.
4. In the **Verification Code** field, enter the code displayed on your authenticator app.
5. Click **Enable**.

Your mobile device is now registered with the IAM service and your account is enabled for MFA. Every time you sign in, you are prompted for your username and password first. After you provide the correct credentials, you will be prompted for a TOTP code generated by the authenticator app on your registered mobile device. *You must have your registered mobile device available every time you sign in to Oracle Cloud Infrastructure.*

To disable MFA for your user account

1. In the upper-right corner of the Console, open the **User** menu () and then select **User Settings**. Your user details are displayed.
2. Click **Disable Multi-Factor Authentication**.
3. Confirm when prompted.

To disable MFA for another user

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Users**.
   A list of the users in your tenancy is displayed.
2. Click the user you want to update.
   The user's details are displayed.

3. Click **Disable Multi-Factor Authentication**.
4. Confirm when prompted.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.



Note

Updates Are Not Immediate Across All Regions

Your IAM resources reside in your home region. To enforce policy across all regions, the IAM service replicates your resources in each region. Whenever you create or change a policy, user, or group, the changes take effect first in the home region, and then are propagated out to your other regions. It can take several minutes for changes to take effect in all regions.

Use these API operations to manage multi-factor authentication devices:

- CreateMfaTotpDevice
- ListMfaTotpDevices
- GetMfaTotpDevice
- DeleteMfaTotpDevice
- ActivateMfaTotpDevice
- GenerateTotpSeed

CHAPTER 14 Key Management

This chapter explains how to create key vaults and encryption keys and how to manage and use them.

Overview of Key Management

Oracle Cloud Infrastructure Key Management provides you with centralized management of the encryption of your data. You can use Key Management to create master encryption keys and data encryption keys, rotate keys to generate new cryptographic material, enable or disable keys for use in cryptographic operations, assign keys to resources, and use keys for encryption and decryption.

Oracle Cloud Infrastructure Object Storage and Oracle Cloud Infrastructure Block Volume integrate with Key Management to support encryption of data in buckets and block or boot volumes. Integration with Oracle Cloud Infrastructure Identity and Access Management (IAM) lets you control who and what services can access which keys and what they can do with those keys. Oracle Cloud Infrastructure Audit integration gives you a way to monitor key usage. Audit tracks administrative actions on keys and vaults.

Keys are stored on highly available and durable hardware security modules (HSM) that meet Federal Information Processing Standards (FIPS) 140-2 Security Level 3 security certification. Key Management uses the Advanced Encryption Standard (AES) as its encryption algorithm and its keys are AES symmetric keys.

Key Management Concepts

The following concepts are integral to understanding Key Management.

KEYS

Keys are logical entities that represent one or more key versions that contain the cryptographic material used to encrypt and decrypt data, protecting the data where it is stored. When processed as part of an encryption algorithm, a key specifies how to

transform plaintext into ciphertext during encryption and how to transform ciphertext into plaintext during decryption. Conceptually, Key Management recognizes two types of encryption keys. You can create master encryption keys using the Console or API. Key Management stores those keys in a key vault. After you have a master encryption key, you can then use the API to generate data encryption keys that the service returns to you. Key Management introduces master encryption keys as an Oracle Cloud Infrastructure resource.

VAULTS

Key vaults are logical entities where Key Management creates and durably stores your keys. Vaults are partitions on a hardware security module that are isolated from one another to ensure the security and integrity of the encryption keys that are stored on them. The type of vault you have determines features and functionality such as degrees of storage isolation, access to management and encryption, scalability, and pricing. At this time, the only type of vault you can create is a virtual private vault. Key Management designates vaults as an Oracle Cloud Infrastructure resource.

KEY VERSIONS

Each master encryption key is automatically assigned a key version. When you rotate a key, Key Management generates a new key version. Periodically rotating keys limits the amount of data encrypted by one key version. Key rotation thereby reduces the risk if a key is ever compromised. A key’s unique, Oracle-assigned identifier, called an Oracle Cloud ID (OCID), remains the same across rotations, but the key version enables Key Management to seamlessly rotate keys to meet any compliance requirements you might have. Although you can't use an older key version for encryption after you rotate it, the key version remains available to decrypt any data that it previously encrypted. Key Management removes the need for you to track which key version was used to encrypt what data because the key's ciphertext contains the information that Key Management requires for decryption.

HARDWARE SECURITY MODULES

When you create a master encryption key using the Console or API, Key Management stores the key version within a hardware security module (HSM) to provide a layer of

physical security. Any given key version, after it's created, is replicated within the service infrastructure as a measure of protection against hardware failures. Key versions are not otherwise stored anywhere else and cannot be exported from an HSM. Key Management uses HSMs that meet Federal Information Processing Standards (FIPS) 140-2 Security Level 3 security certification. This means that the HSM hardware is tamper-evident, has physical safeguards for tamper-resistance, requires identity-based authentication, and deletes keys from the device when it detects tampering.

ENVELOPE ENCRYPTION

The data encryption key used to encrypt your data is, itself, encrypted with a master encryption key. This concept is known as envelope encryption. Oracle Cloud Infrastructure services do not have access to the plaintext data without interacting with Key Management and without access to the master encryption key that is protected by Oracle Cloud Infrastructure Identity and Access Management (IAM). For decryption purposes, Object Storage and Block Volume store only the encrypted form of the data encryption key.

Regions and Availability Domains

You can use Key Management in the ap-seoul-1, ap-tokyo-1, ca-toronto-1, us-ashburn-1, us-phoenix-1, eu-frankfurt-1, and uk-london-1 regions. Unlike other Oracle Cloud Infrastructure services, however, Key Management does not have one regional endpoint for all API operations. The service has one regional endpoint for the provisioning service that handles create, update, and list operations for vaults. For create, update, and list operations for keys, service endpoints are distributed across multiple independent clusters.

Because Key Management has public endpoints, you can directly use data encryption keys generated by Key Management for cryptographic operations in your applications. However, if you want to use master encryption keys with a service that has integrated with Key Management, you can do so only when the service and the key vault that holds the key both exist within the same region.

Key Management maintains copies of encryption keys across all availability domains within a region. This replication makes it possible for Key Management to generate keys even when an availability domain is unavailable.

Private Access to Key Management

Key Management supports private access from Oracle Cloud Infrastructure resources in a virtual cloud network (VCN) through a service gateway. Setting up and using a service gateway on a VCN lets resources (such as the instances that your encrypted volumes are attached to) access public Oracle Cloud Infrastructure services such as Key Management without exposing them to the public internet. No internet gateway is required and resources can be in a private subnet and use only private IP addresses. For more information, see Access to Oracle Services: Service Gateway.

Resource Identifiers

Key Management introduces keys and vaults as Oracle Cloud Infrastructure resources. Most types of Oracle Cloud Infrastructure resources have a unique, Oracle-assigned identifier called an Oracle Cloud ID (OCID). For information about the OCID format and other ways to identify your resources, see Resource Identifiers.

Ways to Access Oracle Cloud Infrastructure

You can access Oracle Cloud Infrastructure using the Console (a browser-based interface) or the REST API. Instructions for the Console and API are included in topics throughout this guide. For a list of available SDKs, see Software Development Kits and Command Line Interface. Terraform does not currently support Key Management.

To access the Console, you must use a supported browser. You can use the **Console** link at the top of this page to go to the sign-in page. You will be prompted to enter your cloud tenant, your user name, and your password.

For general information about using the API, see REST APIs.

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

Limits on Key Management Resources

See Service Limits for a list of applicable limits and instructions for requesting a limit increase.

Managing Keys

This topic describes what you can do with keys and key versions in terms of managing their creation and usage. For information about how you can use keys in cryptographic operations, see Using Keys. For information about what you can do with vaults where you store keys, see Managing Vaults.

Management of keys includes the ability to do the following:

- Create keys
- View key details
- View a list of keys
- View a list of key versions for a specific key
- Update a key name
- Manage a key's tags
- Enable keys for use in cryptographic operations

- Rotate keys to generate new cryptographic material
- Disable keys to prevent their usage in cryptographic operations
- Assign keys to specific resources
- Remove keys from their assignment to specific resources

Required IAM Policy



Warning

Keys associated with volumes and buckets will not work unless you authorize Oracle Cloud Infrastructure Block Volume and Oracle Cloud Infrastructure Object Storage to use keys on your behalf. Additionally, you must also authorize users to delegate key usage to these services in the first place. For more information, see Let a user group delegate key usage in a compartment and Let Block Volume and Object Storage services encrypt and decrypt volumes and buckets in Common Policies.

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: For typical policies that give access to keys and vaults, see Let security admins manage vaults and keys.

Also, be aware that a policy statement with `inspect vaults` gives the specified group the ability to see *all* information about the vaults. Likewise, a policy statement with `inspect keys` gives the specified group the ability to see *all* information about the keys. For more information, see Details for the Key Management Service.

If you're new to policies, see Getting Started with Policies and Common Policies.

Tagging Resources

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

Using the Console

To create a new key

1. Open the navigation menu. Under the **Governance and Administration** group, go to **Security** and click **Key Management**.
2. Under **List Scope**, in the **Compartment** list, click the name of the compartment where you want to create a key.
3. From the list of vaults in the compartment, do one of the following:
   - Click the name of the vault where you want to create a key.
   - Create a new vault for the key by following the instructions in To create a new vault, and then click the name of the vault.
4. Click **Keys**, and then click **Create Key**.
5. In the **Create Key** dialog box, choose a compartment from the **Create in Compartment** list. (Keys can exist outside the compartment the vault is in.)
6. Click **Name**, and then enter a name to identify the key. Avoid entering any confidential information in this field.
7. Specify the key length, in bits, by choosing a length from the **Key Shape: Length** list.
8. Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
9. When you are finished, click **Create Key**.

To view key details

1. Open the navigation menu. Under the **Governance and Administration** group, go to **Security** and click **Key Management**.
2. Under **List Scope**, in the **Compartment** list, click the name of the compartment that contains the vault with the key you're interested in.
3. From the list of vaults in the compartment, click the vault name.
4. Click **Keys**, and then click the name of the key for which you want to see configuration details.
5. The console displays the following information:
    - **OCID:** The unique, Oracle-assigned ID of the key.
    - **Created:** The date and time when you initially created the key.
    - **Compartment:** The unique, Oracle-assigned ID of the compartment that contains the vault that contains the key.
    - **Vault:** The unique, Oracle-assigned ID of the vault that contains the key.
    - **Key Version:** The unique, Oracle-assigned ID of the key version.
    - **Algorithm:** The encryption algorithm used by the key.
    - **Length:** The number of bits in the key length.

To view a list of keys

1. Open the navigation menu. Under the **Governance and Administration** group, go to **Security** and click **Key Management**.
2. Under **List Scope**, in the **Compartment** list, click the name of the compartment that contains the vault with the keys you're interested in.
3. From the list of vaults in the compartment, click the vault name.
4. To see a list of keys in this vault, click **Keys**.

To view a list of key versions

1. Open the navigation menu. Under the **Governance and Administration** group, go to **Security** and click **Key Management**.
2. Under **List Scope**, in the **Compartment** list, click the name of the compartment that contains the vault with the key you're interested in.
3. From the list of vaults in the compartment, click the vault name.
4. Click **Keys**, click the name of the key for which you want to see a list of key versions, and then click **Versions**.

To change the name of a key

1. Open the navigation menu. Under the **Governance and Administration** group, go to **Security** and click **Key Management**.
2. Under **List Scope**, in the **Compartment** list, click the name of the compartment that contains the vault with the key you want to rename.
3. From the list of vaults in the compartment, click the vault name.
4. Click **Keys**, locate the key you want to rename, and then click the Actions icon (three dots) for that key.
5. In the **Actions** menu, click **Edit Name**.
6. In the **Edit Key Name** dialog box, click **Name**, and then enter a new name. Avoid entering any confidential information in this field.
7. When you are finished, click **Update**.

To manage a key's tags

1. Open the navigation menu. Under the **Governance and Administration** group, go to **Security** and click **Key Management**.
2. Under **List Scope**, in the **Compartment** list, click the name of the compartment that contains the vault with the key for which you want to manage tags.

3. From the list of vaults in the compartment, click the vault name.
4. Click **Keys**, locate the key you want to manage, and then click the key name.
5. On the **Key Details** page, click the **Tags** tab to view or edit existing tags. Or, click **Add Tag(s)** to add new ones.

To enable a key

1. Open the navigation menu. Under the **Governance and Administration** group, go to **Security** and click **Key Management**.
2. Under **List Scope**, in the **Compartment** list, click the name of the compartment that contains the vault with the key you want to enable.
3. From the list of vaults in the compartment, click the vault name.
4. Click **Keys**, locate the key you want to enable, and then select the check box next to the key name.
5. In the **Actions** menu, click **Enable**.

To rotate a key

1. Open the navigation menu. Under the **Governance and Administration** group, go to **Security** and click **Key Management**.
2. Under **List Scope**, in the **Compartment** list, click the name of the compartment that contains the vault with the key you want to rotate.
3. From the list of vaults in the compartment, click the vault name.
4. Click **Keys**, locate the key you want to rotate, and then select the check box next to the key name.
5. In the **Actions** menu, click **Rotate Key**. (You can only rotate keys in an enabled state.)
6. In the **Confirm** dialog box, click **Rotate Key**.

To disable a key

1. Open the navigation menu. Under the **Governance and Administration** group, go to **Security** and click **Key Management**.
2. Under **List Scope**, in the **Compartment** list, click the name of the compartment that contains the vault with the key you want to disable.
3. From the list of vaults in the compartment, click the vault name.
4. Click **Keys**, locate the key you want to disable, and then click the Actions icon (three dots) for that key.
5. In the **Actions** menu, click **Disable**.

To assign a key to a new Object Storage bucket

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
2. Under **List Scope**, in the **Compartment** list, choose the compartment where you want to create a bucket that's encrypted with a Key Management master encryption key.
3. Click **Create Bucket**, and then follow the instructions in To create a bucket in Managing Buckets.

To assign a key to an existing Object Storage bucket

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
2. Under **List Scope**, in the **Compartment** list, choose the compartment that contains the bucket that you want to encrypt with a Key Management master encryption key.
3. From the list of buckets, click the bucket name.
4. Do one of the following:
    - If the bucket already has a key assigned to it, next to **Encryption Key**, click **Edit** to assign a different key.

- If the bucket does not already have a key assigned to it, next to **Encryption Key**, click **Assign**.

5. Choose the vault compartment, vault, key compartment, and key.
6. When you are finished, click **Assign** or **Update**, as appropriate.

To assign a key to a new Block Volume

1. Open the navigation menu. Under **Core Infrastructure**, go to **Block Storage** and click **Block Volumes**.
2. Under **List Scope**, in the **Compartment** list, choose the compartment where you want to create a block volume that's encrypted with a Key Management master encryption key.
3. Click **Create Block Volume**, and then follow the instructions in Creating a Volume.

To assign a key to an existing Block Volume

1. Open the navigation menu. Under **Core Infrastructure**, go to **Block Storage** and click **Block Volumes**.
2. Under **List Scope**, in the **Compartment** list, choose the compartment that contains the block volume that you want to encrypt with a Key Management master encryption key.
3. From the list of volumes, click the volume name.
4. If the volume is currently attached to an instance, click **Detach from Instance**. Follow the instructions in the **Detach Block Volume** dialog box as appropriate, click **Continue Detachment**, and then click **OK**.
5. Then, do one of the following:
   - If the volume already has a key assigned to it, next to **Encryption Key**, click **Edit** to assign a different key.
   - If the volume does not already have a key assigned to it, next to **Encryption Key**, click **Assign**.

6. Choose the vault compartment, vault, key compartment, and key.
7. When you are finished, click **Assign** or **Update**, as appropriate.

To create a Compute instance with an encrypted boot volume



Note

These instructions assume you have already fulfilled the prerequisites for creating an instance.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
2. Under **List Scope**, in the **Compartment** list, choose the compartment where you want to create an instance with a boot volume that's encrypted with a Key Management master encryption key.
3. Click **Create Instance**, and then follow the instructions under Using the Console in Launching an Instance.

To assign a key to an existing boot volume



Note

To assign a key to an existing boot volume, you must first detach the boot volume from any instance. However, you can only detach a boot volume from an instance when the instance is stopped. For more information, see Detaching a Boot Volume and Stopping and Starting an Instance.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Boot Volumes**.
2. Under **List Scope**, in the **Compartment** list, choose the compartment that contains the boot volume that you want to encrypt with a Key Management master encryption key.
3. From the list of volumes, click the volume name.
4. Do one of the following:
   - If the volume already has a key assigned to it, next to **Encryption Key**, click **Edit** to assign a different key.
   - If the volume does not already have a key assigned to it, next to **Encryption Key**, click **Assign**.
5. Choose the vault compartment, vault, key compartment, and key.
6. When you are finished, click **Assign** or **Update**, as appropriate.

To remove a key assignment from a bucket

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
2. Under **List Scope**, in the **Compartment** list, choose the compartment that contains the bucket from which you want to remove a Key Management key assignment.
3. From the list of volumes, click the volume name.
4. Next to **Encryption Key**, click **Unassign**.
5. In the **Confirm** dialog box, click **OK** to remove the key assignment from the volume.

To remove a key assignment from a Block Volume

1. Open the navigation menu. Under **Core Infrastructure**, go to **Block Storage** and click **Block Volumes**.
2. Under **List Scope**, in the **Compartment** list, choose the compartment that contains the block volume from which you want to remove a Key Management key assignment.

3. From the list of volumes, click the volume name.
4. Next to **Encryption Key**, click **Unassign**.
5. In the **Confirm** dialog box, click **OK** to remove the key assignment from the volume.

To remove a key assignment from a boot volume

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Boot Volumes**.
2. Under **List Scope**, in the **Compartment** list, choose the compartment that contains the boot volume from which you want to remove a Key Management key assignment.
3. From the list of volumes, click the volume name.
4. Next to **Encryption Key**, click **Unassign**.
5. In the **Confirm** dialog box, click **OK** to remove the key assignment from the volume.

Using the Command Line Interface (CLI)

For information about using the CLI, see Command Line Interface (CLI). For a complete list of flags and options available for CLI commands, see CLI Help.



Tip

Each vault has a unique endpoint for create, update, and list operations for keys. This endpoint is referred to as the control plane URL or management endpoint. Each vault also has a unique endpoint for cryptographic operations. This endpoint is known as the data plane URL or the cryptographic endpoint. When using the CLI for key operations, you must provide the appropriate endpoint for the type of operation. To retrieve a vault's endpoints, see instructions in To view vault configuration details.

To create a new key

Open a command prompt and run `oci kms management key create` to create a new key:

```
oci kms management key create --compartment-id <target_compartment_id> --display-name <key_name> --
key-shape <key_encryption_information> --endpoint <control_plane_url>
```

For example, on a MacOS or Linux machine:

```
oci kms management key create --compartment-id
ocid1.compartment.oc1..example1example25qrlpo4agcmothkbgqgmuz2zzum45ibplooqtabwk3zz --display-name key-1
--key-shape '{"algorithm":"AES","length":"16"}' --endpoint https://exampleaaacu2-management.kms.us-
ashburn-1.oraclecloud.com
```

Or, for example, on a Windows machine:

```
oci kms management key create --compartment-id
ocid1.compartment.oc1..example1example25qrlpo4agcmothkbgqgmuz2zzum45ibplooqtabwk3zz --display-name key-1
--key-shape '{\"algorithm\":\"AES\",\"length\":\"16\"}' --endpoint https://exampleaaacu2-
management.kms.us-ashburn-1.oraclecloud.com
```



Warning

Avoid entering confidential information in the key name.

To create a new key with resource tags

Open a command prompt and run `oci kms management key create` with one or both of the `--defined-tags` and `--freeform-tags` options to create a new key with resource tags:

```
oci kms management key create --compartment-id <target_compartment_id> --display-name <key_name> --
key-shape <JSON_formatted_key_encryption_information> --defined-tags <JSON_formatted_defined_tag> --
freeform-tags <JSON_formatted_freeform_tag> --endpoint <control_plane_url>
```

For example, on a MacOS or Linux machine:

```
oci kms management key create --compartment-id
ocid1.compartment.oc1..example1example25qrlpo4agcmothkbgqgmuz2zzum45ibplooqtabwk3zz --display-name key-1
--key-shape '{"algorithm":"AES","length":"16"}' --defined-tags '{"Operations": {"CostCenter":"42"}}' --
freeform-tags '{"Department":"Finance"}' --endpoint https://exampleaaacu2-management.kms.us-ashburn-
1.oraclecloud.com
```

Or, for example, on a Windows machine:

```
oci kms management key create --compartment-id
ocid1.compartment.oc1..example1example25qrlpo4agcmothkbgqgmuz2zzum45ibplooqtabwk3zz --display-name key-1
--key-shape '{\"algorithm\":\"AES\",\"length\":\"16\"}' --defined-tags '{\"Operations\":
{\"CostCenter\":\"42\"}}' --freeform-tags '{\"Department\":\"Finance\"}' --endpoint
https://exampleaaacu2-management.kms.us-ashburn-1.oraclecloud.com
```



Warning

Avoid entering confidential information in the key name.

To view a key's details

Open a command prompt and run `oci kms management key get` to view a specific key's details:

```
oci kms management key get --key-id <key_OCID> --endpoint <control_plane_url>
```

For example:

```
oci kms management key get --key-id
ocid1.key.region1.sea.exampleaaacu2.examplesmtpsuqmoy4m5cvblugmizcoeu2nfc6b3zfaux2lmqz245gezevsq --
endpoint https://exampleaaacu2-management.kms.us-ashburn-1.oraclecloud.com
```

To view a list of keys

Open a command prompt and run `oci kms management key list` to list keys in a vault:

```
oci kms management key list --compartment-id <target_compartment_id> --endpoint <control_plane_url>
```

For example:

```
oci kms management key list --compartment-id
ocid1.compartment.oc1..example1example25qrlpo4agcmothkbgqgmuz2zzum45ibplooqtabwk3zz --endpoint
https://exampleaaacu2-management.kms.us-ashburn-1.oraclecloud.com
```

To view a list of key versions

Open a command prompt and run `oci kms management key-version list` to view a list of key versions for a specific key:

```
oci kms management key-version list --key-id <key_OCID> --endpoint <control_plane_url>
```

For example:

```
oci kms management key-version list --key-id
ocid1.key.region1.sea.exampleaaacu2.examplesmtpsuqmoy4m5cvblugmizcoeu2nfc6b3zfaux2lmqz245gezevsq --
endpoint https://exampleaaacu2-management.kms.us-ashburn-1.oraclecloud.com
```

To change the name of a key

Open a command prompt and run `oci kms management key update` to edit a key's name.



Warning

Avoid entering confidential information in the key name.

```
oci kms management key update --key-id <key_OCID> --display-name <new_key_name> --endpoint <control_plane_url>
```

For example:

```
oci kms management key update --key-id
ocid1.key.region1.sea.exampleaaacu2.examplesmtpsuqmoy4m5cvblugmizcoeu2nfc6b3zfaux2lmqz245gezevsq --
display-name key-A --endpoint https://exampleaaacu2-management.kms.us-ashburn-1.oraclecloud.com
```

To enable a key

Open a command prompt and run `oci kms management key enable` to enable a key:

```
oci kms management key enable --key-id <target_key_id> --endpoint <control_plane_url>
```

For example:

```
oci kms management key enable --key-id
ocid1.key.region1.sea.exampleaaacu2.examplesmtpsuqmoy4m5cvblugmizcoeu2nfc6b3zfaux2lmqz245gezevsq --
endpoint https://exampleaaacu2-management.kms.us-ashburn-1.oraclecloud.com
```

To rotate a key

Open a command prompt and run `oci kms management key rotate` to rotate a key:

```
oci kms management key rotate --key-id <target_key_id> --endpoint <control_plane_url>
```

For example:

```
oci kms management key rotate --key-id
ocid1.key.region1.sea.exampleaaacu2.examplesmtpsuqmoy4m5cvblugmizcoeu2nfc6b3zfaux2lmqz245gezevsq --
endpoint https://exampleaaacu2-management.kms.us-ashburn-1.oraclecloud.com
```

To disable a key

Open a command prompt and run `oci kms management key disable` to disable a key:

```
oci kms management key disable --key-id <target_key_id> --endpoint <control_plane_url>
```

For example:

```
oci kms management key disable --key-id
ocid1.key.region1.sea.exampleaaacu2.examplesmtpsuqmoy4m5cvblugmizcoeu2nfc6b3zfaux2lmqz245gezevsq --
endpoint https://exampleaaacu2-management.kms.us-ashburn-1.oraclecloud.com
```

To assign a key to an Object Storage bucket

Open a command prompt and run `oci os bucket create` to create a bucket that is encrypted with a Key Management key:

```
oci os bucket create --name <bucket_name> --compartment-id <target_compartment_id> --kms-key-id
<target_key_id>
```

For example:

```
oci os bucket create --name Bucket-1 --compartment-id
ocid1.compartment.oc1..example1example25qrlpo4agcmothkbgqgmuz2zzum45ibplooqtabwk3zz --kms-key-id
ocid1.key.region1.sea.exampleaaacu2.examplesmtpsuqmoy4m5cvblugmizcoeu2nfc6b3zfaux2lmqz245gezevsq --
namespace-name example_namespace
```



Warning

Avoid entering confidential information in the bucket name.

To update the key assigned to an Object Storage bucket

Open a command prompt and run `oci os bucket update` to update the Key Management key assigned to a bucket:

```
oci os bucket update --name <bucket_name> --namespace-name <your_namespace> --kms-key-id <target_key_id>
```

For example:

```
oci os bucket update --name Bucket-1 --namespace-name example_namespace --kms-key-id
ocid1.key.region1.sea.exampleaaacu2.examplesmtpsuqmoy4m5cvblugmizcoeu2nfc6b3zfaux2lmqz245gezevsq
```

To create a block volume that's encrypted with a Key Management key

Open a command prompt and run `oci bv volume create` to create a block volume that is encrypted with a Key Management key:

```
oci bv volume create --display-name <volume_name> --compartment-id <target_compartment_id> --size-in-gbs <volume_size> --availability-domain <target_availability_domain> --kms-key-id <target_key_id>
```

For example:

```
oci bv volume create --display-name EncryptedBlockVolume --compartment-id
ocid1.compartment.oc1..example1example25qrlpo4agcmothkbgqgmuz2zzum45ibplooqtabwk3zz --size-in-gbs 50 --
availability-domain AAbC:US-ASHBURN-AD-1 --kms-key-id
ocid1.key.region1.sea.exampleaaacu2.examplesmtpsuqmoy4m5cvblugmizcoeu2nfc6b3zfaux2lmqz245gezevsq
```



Warning

Avoid entering confidential information in the volume name.

To update a key assigned to an existing Block Volume



Tip

If the volume is currently attached to an instance, you must first detach it. To do so, open a command prompt and run `oci compute volume-attachment detach --volume-attachment-id <target_blockvolume-attachment_id>`. For more information, see Oracle Cloud Infrastructure CLI Command Reference.

Open a command prompt and run `oci bv volume-kms-key update` to assign a new Key Management key to an existing block volume:

```
oci bv volume-kms-key update --volume-id <target_blockvolume_id> --kms-key-id <new_key_id>
```

For example:

```
oci bv volume-kms-key update --volume-id
ocid1.volume.oc1.sea.examplerwzq7bnohn5vf6b7k4zkp54miqfcvg6xsuvkllgzzw63mfuu6z5fa --kms-key-id
ocid1.key.region1.sea.exampleaaacu2.examplesmtpsuqmoy4m5cvblugmizcoeu2nfc6b3zfaux2lmqz245gezevsq
```

To create a boot volume that's encrypted with a Key Management key

Open a command prompt and run `oci bv boot-volume create` to create a boot volume that is encrypted with a Key Management key:

```
oci bv boot-volume create --display-name <volume_name> --compartment-id <target_compartment_id> --
size-in-gbs <volume_size> --availability-domain <target_availability_domain> --kms-key-id <target_key_
id>
```

For example:

```
oci bv boot-volume create --display-name EncryptedBlockVolume --compartment-id
ocid1.compartment.oc1..example1example25qrlpo4agcmothkbgqgmuz2zzum45ibplooqtabwk3zz --size-in-gbs 50 --
```

```
availability-domain AAbC:US-ASHBURN-AD-1 --kms-key-id
ocid1.key.region1.sea.exampleaaacu2.examplesmtpsuqmoy4m5cvblugmizcoeu2nfc6b3zfaux2lmqz245gezevsq
```



Warning

Avoid entering confidential information in the volume name.

To create a Compute instance with a boot volume that's encrypted with a Key Management key



Note

These instructions assume you have already fulfilled the prerequisites for creating an instance.

1. First, create the JSON input for configuring the instance and boot volume: Open a command prompt and run `oci compute instance launch --generate-full-command-json-input`.
2. Copy, and then paste the output from the command into a text file for editing. Edit the JSON to provide values appropriate for your tenancy and desired image operating system and instance shape. The following example shows the minimum settings required to create an instance and encrypted boot volume.

```
{
"availabilityDomain": "ABcD:US-ASHBURN-AD-1",
"compartmentId":
"ocid1.tenancy.oc1..examplea54hlbsiugecvb4g67tnth7ouk4iivkpysfauxcetd55uiunrykhq",
"displayName": "InstanceWithEncryptedBootVolume",
"metadata": {
},
"shape": "VM.Standard1.1",
```

```
"subnetId": "ocid1.subnet.oc1.iad.exampleaurihk3x3yl2vcvb53uz22zgauoujtcwvtbxvfauxdvsjmdfv4dza",
"sourceDetails": {
   "sourceType": "image",
   "imageId": "ocid1.image.oc1.iad.exampleaeookczfwutjxzcvb2gcdgdx4yk6xls7d5fhtlfauxzpaxdedny4a",
   "kmsKeyId":
"ocid1.key.oc1.iad.exampleoaaeug.examplera4soq2vescvbjmwredhewtto7rlfauxhvme73y7jayxx6rpaenlq"
   }
}
```



Warning

Avoid entering confidential information in the instance name.

3. Save the file with a ".json" file extension.
4. In the command prompt, run `oci compute instance launch --from-json file://<file_path>`, providing the location of the file you saved in the previous step. For example:

```
oci compute instance launch --from-json file://c:\temp\compute-boot-volume.json
```

To update a key assigned to an existing boot volume



Tip

If the volume is currently attached to an instance, you must first detach the volume. To do so, you must first stop the instance. To stop an instance, open a command prompt and run `oci compute instance action --instance-id <target_instance_id> --action STOP`. Then, to detach the boot volume, run `oci compute`



`boot-volume-attachment detach --boot-volume-attachment-id <target_bootvolume-attachment_id>`. For more information, see the Oracle Cloud Infrastructure CLI Command Reference.

Open a command prompt and run `oci bv boot-volume-kms-key update` to assign a new Key Management key to an existing boot volume:

```
oci bv boot-volume-kms-key update --boot-volume-id <target_bootvolume_id> --kms-key-id <new_key_id>
```

For example:

```
oci bv boot-volume-kms-key update --boot-volume-id
ocid1.bootvolume.oc1.sea.examples6hvjs6j6mqwcdv4gfzhtanon3fsqyviqeh522be6wv7x7abz7pq --kms-key-id
ocid1.key.region1.sea.exampleaaacu2.examplesmtpsuqmoy4m5cvblugmizcoeu2nfc6b3zfaux2lmqz245gezevsq
```

To remove the key assigned to an Object Storage bucket

Open a command prompt and run `oci os bucket update` to remove the Key Management key assigned to a bucket:

```
oci os bucket update --name <bucket_name> --namespace-name <your_namespace> --kms-key-id ""
```

For example:

```
oci os bucket update --name Bucket-1 --kms-key-id "" --namespace-name example_namespace
```

To remove a key assigned to a Block Volume

Open a command prompt and run `oci bv volume-kms-key delete` to remove the Key Management key assigned to an existing block volume:

```
oci bv volume-kms-key delete --volume-id <target_blockvolume_id>
```

For example:

```
oci bv volume-kms-key delete --volume-id
ocid1.volume.oc1.sea.examplerwzq7bnohn5vf6b7k4zkp54miqfcvg6xsuvkllgzzw63mfuu6z5fa
```

To remove a key assigned to a Block Volume boot volume

Open a command prompt and run `oci bv boot-volume-kms-key delete` to remove the Key Management key assigned to an existing boot volume:

```
oci bv boot-volume-kms-key delete --boot-volume-id <target_bootvolume_id>
```

For example:

```
oci bv boot-volume-kms-key delete --boot-volume-id
ocid1.bootvolume.oc1.sea.exampless6hvjs6j6mqwcdv4gfzhtanon3fsqyviqeh522be6wv7x7abz7pq
```

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operations to manage keys:

- CreateKey
- DisableKey
- EnableKey
- GetKey
- UpdateKey
- CreateKeyVersion
- GetKeyVersion
- ListKeys
- ListKeyVersions

Managing Vaults

This topic describes what you can do with vaults. For information about what you can do with keys, see Managing Keys.

Key Management lets you create virtual private vaults in your tenancy. A virtual private vault provides you with a dedicated partition in a hardware security module (HSM), offering a level of storage isolation for encryption keys that’s effectively equivalent to a virtual independent HSM. As such, virtual private vaults have dedicated administration and users. They also restrict other tenants from accessing your encryption keys.

Vault management tasks include the following:

- Creating a vault
- Viewing vault configuration details
- Updating the vault name
- Managing vault tags
- Deleting a vault

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: For typical policies that give access to keys and vaults, see Let security admins manage vaults and keys.

Also, be aware that a policy statement with `inspect vaults` gives the specified group the ability to see *all* information about the vaults. Likewise, a policy statement with `inspect keys` gives the specified group the ability to see *all* information about the keys. For more information, see Details for the Key Management Service.

If you're new to policies, see Getting Started with Policies and Common Policies.

Tagging Resources

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

Using the Console

To view vault configuration details

1. Open the navigation menu. Under the **Governance and Administration** group, go to **Security** and click **Key Management**.
2. Under **List Scope**, in the **Compartment** list, click the name of the compartment that contains the vault you want to view.
3. From the list of vaults in the compartment, click the name of the vault.
4. The console displays the following information:
    - **OCID:** The unique, Oracle-assigned ID of the vault.
    - **Created:** The date and time when you initially created the vault.
    - **Compartment:** The unique, Oracle-assigned ID of the compartment that contains the vault.
    - **Vault Type:** The type of vault. At this time, you can only have a virtual private vault (VIRTUAL_PRIVATE).
    - **Control Plane URL:** The service endpoint for CreateKey, CreateKeyVersion, EnableKey, DisableKey, UpdateKey, ListKeys, ListKeyVersions, GetKey, and GetKeyVersion operations.
    - **Data Plane URL:** The service endpoint for Encrypt, Decrypt, and GenerateDataEncryptionKey operations.

To create a new vault

1. Open the navigation menu. Under the **Governance and Administration** group, go to **Security** and click **Key Management**.
2. Under **List Scope**, in the **Compartment** list, click the name of the compartment where you want to create the vault.
3. Click **Create Vault**.
4. In the **Create Vault** dialog box, click **Name**, and then enter a display name for the vault. Avoid entering any confidential information in this field.
5. Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
6. When you are finished, click **Create**.

To change a vault name

1. Open the navigation menu. Under the **Governance and Administration** group, go to **Security** and click **Key Management**.
2. Under **List Scope**, in the **Compartment** list, click the name of the compartment that contains the vault you want to rename.
3. From the list of vaults in the compartment, click the name of the vault.
4. On the **Vault Details** page, click **Edit Name**.
5. In the **Edit Vault Name** dialog box, click **Name**, and then enter a new display name for the vault. Avoid entering any confidential information in this field.
6. When you are finished, click **Save**.

To manage a vault's tags

1. Open the navigation menu. Under the **Governance and Administration** group, go to **Security** and click **Key Management**.
2. Under **List Scope**, in the **Compartment** list, click the name of the compartment that contains the vault for which you want to manage tags.
3. From the list of vaults in the compartment, click the name of the vault.
4. On the **Vault Details** page, click the **Tags** tab to view or editing existing tags. Or, click **Add Tag(s)** to add new ones.

To delete a vault



Note

When you delete a vault, the vault and all its associated keys go into a pending deletion state until the waiting period expires. By default, this is 30 days, but can be set from a minimum of 7 days up to a maximum of 30 days. When a vault is deleted, all its associated keys are also deleted.

1. Open the navigation menu. Under the **Governance and Administration** group, go to **Security** and click **Key Management**.
2. Under **List Scope**, in the **Compartment** list, click the name of the compartment that contains the vault you want to delete.
3. From the list of vaults in the compartment, click the name of the vault.
4. On the **Vault Details** page, click **Delete**.
5. To confirm that you want to delete the vault, type the name of the vault, and then

choose the date and time you want the vault to be deleted.

6. When you are finished, click **Delete Vault**.

To cancel the deletion of a vault

1. Open the navigation menu. Under the **Governance and Administration** group, go to **Security** and click **Key Management**.
2. Under **List Scope**, in the **Compartment** list, click the name of the compartment that contains the vault that's in a pending deletion state.
3. From the list of vaults in the compartment, click the name of the vault.
4. On the **Vault Details** page, click **Cancel Deletion**.
5. To confirm that you want to cancel deletion of the vault, click **Cancel Deletion**.

Using the Command Line Interface (CLI)

For information about using the CLI, see Command Line Interface (CLI). For a complete list of flags and options available for CLI commands, see CLI Help.

To view vault configuration details

Open a command prompt and run `oci kms management vault get` to view the configuration details for a vault:

```
oci kms management vault get --vault-id <target_vault_id>
```

For example:

```
oci kms management vault get --vault-id
ocid1.vault.region1.sea.exampleaaacu2.examplesrcvbtqe5wgrxn2jua3olmeausn5fauxseubwu5my5tf3w3j33edq
```

To create a new vault

Open a command prompt and run `oci kms management vault create` to create a new vault:

```
oci kms management vault create --compartment-id <target_compartment_id> --display-name <vault_name> --vault-type VIRTUAL_PRIVATE
```

For example:

```
oci kms management vault create --compartment-id
ocid1.compartment.oc1..example1example25qrlpo4agcmothkbgqgmuz2zzum45ibplooqtabwk3zz --display-name
vault-1 --vault-type VIRTUAL_PRIVATE
```



Warning

Avoid entering confidential information in the vault name.

To create a new vault with resource tags

Open a command prompt and run `oci kms management vault create` with one or both of the `--defined-tags` and `--freeform-tags` options to create a new vault with resource tags:

```
oci kms management vault create --compartment-id <target_compartment_id> --display-name <vault_name> --vault-type VIRTUAL_PRIVATE --defined-tags <JSON_formatted_defined_tag> --freeform-tags <JSON_formatted_freeform_tag>
```

For example, on a MacOS or Linux machine:

```
oci kms management vault create --compartment-id
ocid1.compartment.oc1..example1example25qrlpo4agcmothkbgqgmuz2zzum45ibplooqtabwk3zz --display-name
vault-1 --vault-type VIRTUAL_PRIVATE --defined-tags '{"Operations": {"CostCenter": "42"}}' --freeform-tags '{"Department":"Finance"}'
```

Or, for example, on a Windows machine:

```
oci kms management vault create --compartment-id
ocid1.compartment.oc1..example1example25qrlpo4agcmothkbgqgmuz2zzum45ibplooqtabwk3zz --display-name
vault-1 --vault-type VIRTUAL_PRIVATE --defined-tags '{\"Operations\": {\"CostCenter\":\"42\"}}' --
freeform-tags '{\"Department\":\"Finance\"}'
```



Warning

Avoid entering confidential information in the vault name.

To change a vault name

Open a command prompt and run `oci kms management vault update` to change a vault's name:

```
oci kms management vault update --vault-id <target_vault_id>
```

For example:

```
oci kms management vault update --vault-id
ocid1.vault.region1.sea.exampleaaacu2.examplesrcvbtqe5wgrxn2jua3olmeausn5fauxseubwu5my5tf3w3j33edq --
display-name new-vault-name
```

To delete a vault

Open a command prompt and run `oci kms management vault schedule-deletion` to delete a vault:

```
oci kms management vault schedule-deletion --vault-id <target_vault_id>
```

For example:

```
oci kms management vault schedule-deletion --vault-id
ocid1.vault.region1.sea.exampleaaacu2.examplesrcvbtqe5wgrxn2jua3olmeausn5fauxseubwu5my5tf3w3
```

When you delete a vault, the vault and all its associated keys go into a pending deletion state until the waiting period expires. By default, this is 30 days, but can be set from a minimum of 7 days up to a maximum of 30 days. When a vault is deleted, all its associated keys are also deleted.

To cancel the deletion of a vault

Open a command prompt and run `oci kms management vault cancel-deletion` to cancel the pending deletion of a vault:

```
oci kms management vault cancel-deletion --vault-id <target_vault_id>
```

For example:

```
oci kms management vault cancel-deletion --vault-id
ocid1.vault.region1.sea.exampleaaacu2.examplesrcvbtqe5wgrxn2jua3olmeausn5fauxseubwu5my5tf3w3
```

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operations to manage vaults:

- CancelVaultDeletion
- CreateVault
- GetVault
- ListVaults
- ScheduleVaultDeletion
- UpdateVault

Using Keys

This topic describes what you can do with keys in terms of cryptographic operations. For information about managing keys, see Managing Keys. For information about managing the vaults in which you store keys, see Managing Vaults.

Cryptographic operations include the following:

- Encrypting data
- Decrypting data
- Generating data encryption keys

You can use either the command line interface (CLI) or API to perform cryptographic operations.

Required IAM Policy



Warning

Keys associated with volumes and buckets will not work unless you authorize Oracle Cloud Infrastructure Block Volume and Oracle Cloud Infrastructure Object Storage to use keys on your behalf. Additionally, you must also authorize users to delegate key usage to these services in the first place. For more information, see Let a user group delegate key usage in a compartment and Let Block Volume and Object Storage services encrypt and decrypt volumes and buckets in Common Policies.

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have

permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: For typical policies that give access to keys and vaults, see Let security admins manage vaults and keys.

Also, be aware that a policy statement with `inspect vaults` gives the specified group the ability to see *all* information about the vaults. Likewise, a policy statement with `inspect keys` gives the specified group the ability to see *all* information about the keys. For more information, see Details for the Key Management Service.

If you're new to policies, see Getting Started with Policies and Common Policies.

Using the Command Line Interface (CLI)

For information about using the CLI, see Command Line Interface (CLI). For a complete list of flags and options available for CLI commands, see CLI Help.



Tip

Each vault has a unique endpoint for create, update, and list operations for keys. This endpoint is referred to as the control plane URL or management endpoint. Each vault also has a unique endpoint for cryptographic operations. This endpoint is known as the data plane URL or the cryptographic endpoint. When using the CLI for key operations, you must provide the appropriate endpoint for the type of operation. To retrieve a vault's endpoints, see instructions in To view vault configuration details.

To encrypt data by using your Key Management master encryption key

Open a command prompt and run `oci kms crypto encrypt` to encrypt data:

```
oci kms crypto encrypt --key-id <key_OCID> --plaintext <base64_string> --endpoint <data_plane_url>
```

For example:

```
oci kms crypto encrypt --key-id
ocid1.key.region1.sea.exampleaaacu2.examplesmtpsuqmoy4m5cvblugmizcoeu2nfc6b3zfaux2lmqz245gezevsq --
plaintext VGhlIHF1aWNrIGJyb3duIGZveCBqdW1wcyBvdmVyIHRoZSBsYXp5IGRvZy4= --endpoint https://exampleaaacu3-
crypto.kms.us-ashburn-1.oraclecloud.com
```

Optionally, you can include the `associated-data` option to provide an encryption context that might contain useful, but non-secret, information about the encrypted data. That information is associated with the encrypted data such that the data cannot be decrypted without it, providing an additional layer of protection. Associated data must be properly formatted JSON.

```
oci kms crypto encrypt --key-id
ocid1.key.region1.sea.exampleaaacu2.examplesmtpsuqmoy4m5cvblugmizcoeu2nfc6b3zfaux2lmqz245gezevsq --
plaintext VGhlIHF1aWNrIGJyb3duIGZveCBqdW1wcyBvdmVyIHRoZSBsYXp5IGRvZy4= --associated-data '
{"CustomerId":"12345", "Custom Data":"custom data"}' --endpoint https://exampleaaacu3-crypto.kms.us-
ashburn-1.oraclecloud.com
```

To decrypt data by using your Key Management master encryption key

Open a command prompt and run `oci kms crypto decrypt` to decrypt data:

```
oci kms crypto decrypt --key-id <key_OCID> --plaintext <base64_string> --endpoint <data_plane_url>
```

For example:

```
oci kms crypto decrypt --key-id
ocid1.key.region1.sea.exampleaaacu2.examplesmtpsuqmoy4m5cvblugmizcoeu2nfc6b3zfaux2lmqz245gezevsq --
plaintext VGhlIHF1aWNrIGJyb3duIGZveCBqdW1wcyBvdmVyIHRoZSBsYXp5IGRvZy4= --endpoint https://exampleaaacu3-
crypto.kms.us-ashburn-1.oraclecloud.com
```

If the data you want to decrypt had an encryption context associated with it at the time of encryption, the same encryption context is required to decrypt the data. For example, the `--associated-data` in the following sample matches what was provided in the preceding sample command for encrypting data.

```
oci kms crypto decrypt --key-id
ocid1.key.region1.sea.exampleaaacu2.examplesmtpsuqmoy4m5cvblugmizcoeu2nfc6b3zfaux2lmqz245gezevsq --
plaintext VGhlIHF1aWNrIGJyb3duIGZveCBqdW1wcyBvdmVyIHRoZSBsYXp5IGRvZy4= --associated-data '
```

```
{"CustomerId":"12345", "Custom Data":"custom data"}' --endpoint https://exampleaaacu3-crypto.kms.us-ashburn-1.oraclecloud.com
```

To generate a data encryption key from your Key Management master encryption key

Open a command prompt and run `oci kms crypto generate-data-encryption-key` to generate a data encryption key that you can then use to encrypt and decrypt data:

```
oci kms crypto generate-data-encryption-key --key-id <key_OCID> --key-shape <key_encryption_information> --include-plaintext-key <Boolean_value> --endpoint <data_plane_url>
```

For example:

```
oci kms crypto generate-data-encryption-key --key-id ocid1.key.region1.sea.exampleaaacu2.examplesmtpsuqmoy4m5cvblugmizcoeu2nfc6b3zfaux2lmqz245gezevsq --key-shape file://path/to/json/file --include-plaintext-key true --endpoint https://exampleaaacu3-crypto.kms.us-ashburn-1.oraclecloud.com
```

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operations to use keys in cryptographic operations:

- Decrypt
- Encrypt
- GenerateDataEncryptionKey

CHAPTER 15 Load Balancing

This chapter explains how to set up a load balancer.

Overview of Load Balancing

The Oracle Cloud Infrastructure Load Balancing service provides automated traffic distribution from one entry point to multiple servers reachable from your virtual cloud network (VCN). The service offers a load balancer with your choice of a public or private IP address, and provisioned bandwidth.

A load balancer improves resource utilization, facilitates scaling, and helps ensure high availability. You can configure multiple load balancing policies and application-specific health checks to ensure that the load balancer directs traffic only to healthy instances. The load balancer can reduce your maintenance window by draining traffic from an unhealthy application server before you remove it from service for maintenance.

How Load Balancing Works

The Load Balancing service enables you to create a public or private load balancer within your VCN. A public load balancer has a public IP address that is accessible from the internet. A private load balancer has an IP address from the hosting subnet, which is visible only within your VCN. You can configure multiple listeners for an IP address to load balance transport Layer 4 and Layer 7 (TCP and HTTP) traffic. Both public and private load balancers can route data traffic to any backend server that is reachable from the VCN.

Public Load Balancer

To accept traffic from the internet, you create a public load balancer. The service assigns it a public IP address that serves as the entry point for incoming traffic. You can associate the public IP address with a friendly DNS name through any DNS vendor.

A public load balancer is regional in scope. If your region includes multiple availability domains, a public load balancer requires either a regional subnet (recommended) or two availability domain-specific (AD-specific) subnets, each in a separate availability domain. With a regional subnet, the Load Balancing service creates a primary load balancer and a standby load balancer, each in a different availability domain, to ensure accessibility even during an availability domain outage. If you create a load balancer in two AD-specific subnets, one subnet hosts the primary load balancer and the other hosts a standby load balancer. If the primary load balancer fails, the public IP address switches to the secondary load balancer. The service treats the two load balancers as equivalent and you cannot specify which one is "primary".

Whether you use regional or AD-specific subnets, each load balancer requires one private IP address from its host subnet. The Load Balancing service supplies a floating public IP address to the primary load balancer. The floating public IP address does not come from your backend subnets.

If your region includes only one availability domain, the service requires just one subnet, either regional or AD-specific, to host both the primary and standby load balancers. The primary and standby load balancers each require a private IP address from the host subnet, in addition to the assigned floating public IP address. If there is an availability domain outage, the load balancer has no failover.



Warning

You cannot specify a private subnet for your public load balancer.

Private Load Balancer

To isolate your load balancer from the internet and simplify your security posture, you can create a private load balancer. The Load Balancing service assigns it a private IP address that serves as the entry point for incoming traffic.

When you create a private load balancer, the service requires only one subnet to host both the

primary and standby load balancers. The load balancer can be regional or AD-specific, depending on the scope of the host subnet. The load balancer is accessible only from within the VCN that contains the host subnet, or as further restricted by your security list rules.

The assigned floating private IP address is local to the host subnet. The primary and standby load balancers each require an additional private IP address from the host subnet.

If there is an availability domain outage, a private load balancer created in a regional subnet within a multi-AD region provides failover capability. A private load balancer created in an AD-specific subnet, or in a regional subnet within a single availability domain region, has no failover capability in response to an availability domain outage.

All Load Balancers

Your load balancer has a backend set to route incoming traffic to your Compute instances. The backend set is a logical entity that includes:

- A list of backend servers.
- A load balancing policy.
- A health check policy.
- Optional SSL handling.
- Optional session persistence configuration.

The backend servers (Compute instances) associated with a backend set can exist anywhere, as long as the associated security lists and route tables allow the intended traffic flow.

Every subnet within your VCN has security lists and a route table. Rules within the security lists determine whether a subnet can accept data traffic from the internet or from another subnet. When you add backend servers to a backend set, the Load Balancing service can suggest appropriate security list rules, or you configure them yourself through the Networking service. See Security Lists for more information.

Oracle recommends that you create your load balancer in a regional subnet.

Oracle recommends that you distribute your backend servers across all availability domains within the region.

To create a minimal system with a functioning load balancer, you must:

- For a public load balancer, create a VCN with an internet gateway and a public regional subnet.



Warning

You cannot specify a private subnet for your public load balancer.

- For a private load balancer, create a VCN with at least one private subnet.
- Create at least two Compute instances, each in a separate availability domain.
- Create a load balancer.
- Create a backend set with a health check policy.
- Add backend servers (Compute instances) to the backend set.
- Create a listener, with optional SSL handling.
- Update the load balancer subnet security list so it allows the intended traffic.



Note

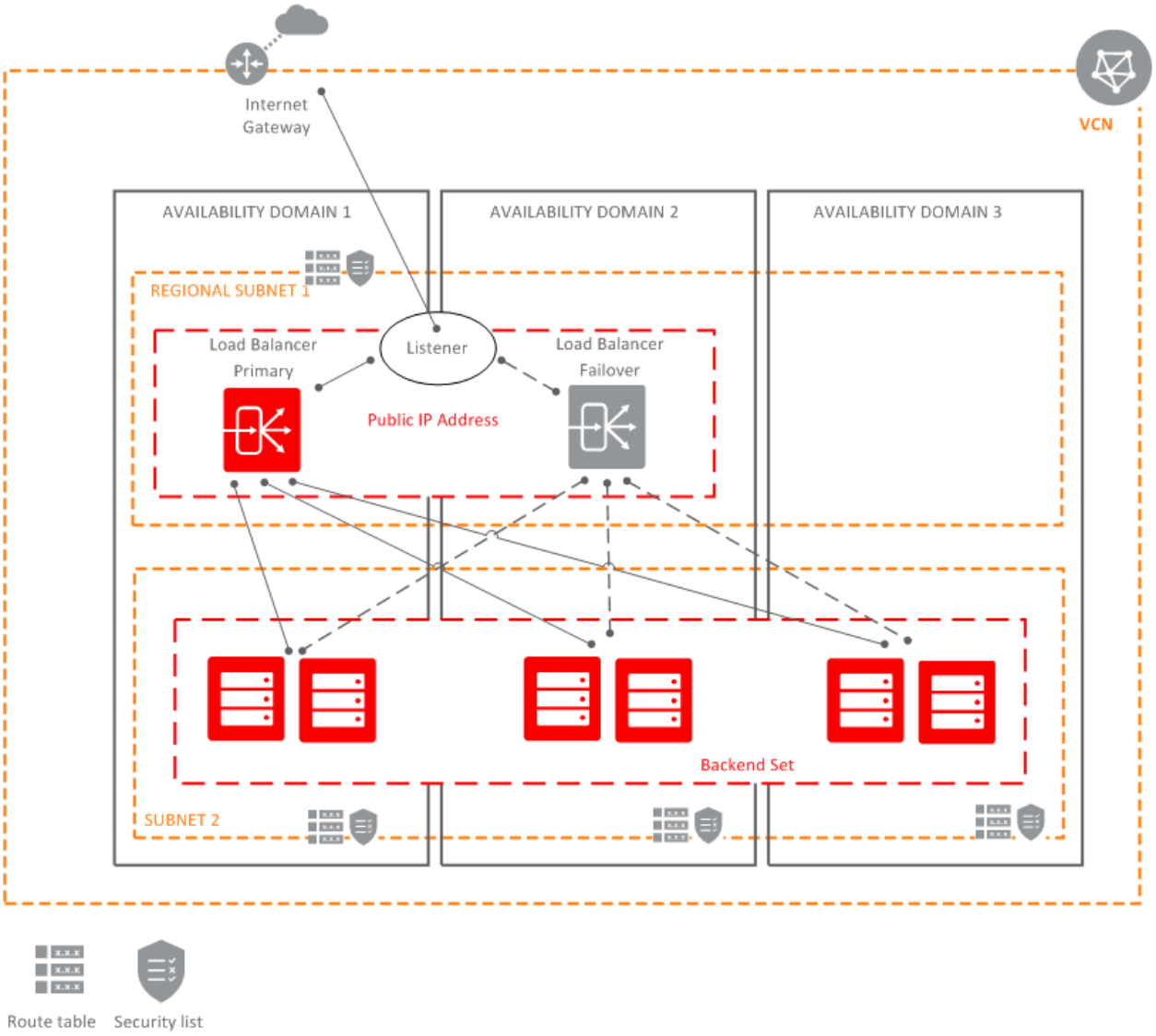
Private IP Address Consumption

A public load balancer created in one public subnet consumes two private IP addresses from the host subnet.

A public load balancer created in two public subnets consumes two private IP addresses, one from each host subnet.

A private load balancer created in a single subnet consumes three private IP addresses from the host subnet.

The following diagram provides a high-level view of a simple public load balancing system configuration. Far more sophisticated and complex configurations are common.



Load Balancing Concepts

The following concepts are essential to working with Load Balancing.

BACKEND SERVER

An application server responsible for generating content in reply to the incoming TCP or HTTP traffic. You typically identify application servers with a unique combination of overlay (private) IPv4 address and port, for example, 10.10.10.1:8080 and 10.10.10.2:8080.

For more information, see Managing Backend Servers.

BACKEND SET

A logical entity defined by a list of backend servers, a load balancing policy, and a health check policy. SSL configuration is optional. The backend set determines how the load balancer directs traffic to the collection of backend servers.

For more information, see Managing Backend Sets.

CERTIFICATES

If you use HTTPS or SSL for your listener, you must associate an SSL server certificate (X.509) with your load balancer. A certificate enables the load balancer to terminate the connection and decrypt incoming requests before passing them to the backend servers.

For more information, see Managing SSL Certificates.

HEALTH CHECK

A test to confirm the availability of backend servers. A health check can be a request or a connection attempt. Based on a time interval you specify, the load balancer applies the health check policy to continuously monitor backend servers. If a server fails the health check, the load balancer takes the server temporarily out of rotation. If the server subsequently passes the health check, the load balancer returns it to the rotation.

You configure your health check policy when you create a backend set. You can configure TCP-level or HTTP-level health checks for your backend servers.

- TCP-level health checks attempt to make a TCP connection with the backend servers and validate the response based on the connection status.
- HTTP-level health checks send requests to the backend servers at a specific URI and validate the response based on the status code or entity data (body) returned.

The service provides application-specific health check capabilities to help you increase availability and reduce your application maintenance window.

For more information on health check configuration, see Editing Health Check Policies.

HEALTH STATUS

An indicator that reports the general health of your load balancers and their components.

For more information, see the **Health Status** section of Editing Health Check Policies.

LISTENER

A logical entity that checks for incoming traffic on the load balancer's IP address. You configure a listener's protocol and port number, and the optional SSL settings. To handle TCP, HTTP, and HTTPS traffic, you must configure multiple listeners.

Supported protocols include:

- TCP
- HTTP/1.0

- HTTP/1.1

For more information, see Managing Load Balancer Listeners.

LOAD BALANCING POLICY

A load balancing policy tells the load balancer how to distribute incoming traffic to the backend servers. Common load balancer policies include:

- Round robin
- Least connections
- IP hash

For more information, see How Load Balancing Policies Work.

PATH ROUTE SET

A set of path route rules to route traffic to the correct backend set without using multiple listeners or load balancers.

For more information, see Managing Request Routing.

REGIONS AND AVAILABILITY DOMAINS

The Load Balancing service manages application traffic across availability domains within a region. A region is a localized geographic area, and an availability domain is one or more data centers located within a region. A region is composed of several availability domains.

For more information, see Regions and Availability Domains.

SESSION PERSISTENCE

A method to direct all requests originating from a single logical client to a single backend web server.

For more information, see Session Persistence.

SHAPE

A template that determines the load balancer's total pre-provisioned maximum capacity (bandwidth) for ingress plus egress traffic. Available shapes include 100 Mbps, 400 Mbps, and 8000 Mbps.



Tip

Pre-provisioned maximum capacity applies to aggregated connections, not to a single client attempting to use the full bandwidth.

SSL

Secure Sockets Layer (SSL) is a security technology for establishing an encrypted link between a client and a server. You can apply the following SSL configurations to your load balancer:

SSL TERMINATION

The load balancer handles incoming SSL traffic and passes the unencrypted request to a backend server.

END TO END SSL

The load balancer terminates the SSL connection with an incoming traffic client, and then initiates an SSL connection to a backend server.

SSL TUNNELING

If you configure the load balancer's listener for TCP traffic, the load balancer tunnels incoming SSL connections to your application servers.

Load Balancing supports the TLS 1.2 protocol with a default setting of **strong** cipher strength. The default supported ciphers include:

- ECDHE-RSA-AES256-GCM-SHA384

- ECDHE-RSA-AES256-SHA384
- ECDHE-RSA-AES128-GCM-SHA256
- ECDHE-RSA-AES128-SHA256
- DHE-RSA-AES256-GCM-SHA384
- DHE-RSA-AES256-SHA256
- DHE-RSA-AES128-GCM-SHA256
- DHE-RSA-AES128-SHA256

For more information, see Managing SSL Certificates.

SUBNET

A subdivision you define in a VCN, such as 10.0.0.0/24 and 10.0.1.0/24. A subnet can span a region or exist within in a single availability domain. A subnet consists of a contiguous range of IP addresses that do not overlap with other subnets in the VCN. For each subnet, you specify the routing rules and security lists that apply to it.

For more information on subnets, see VCNs and Subnets and Public vs. Private Subnets.

TAGS

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

VIRTUAL HOSTNAME

A virtual server name applied to a listener to enhance request routing.

For more information, see Managing Request Routing.

VIRTUAL CLOUD NETWORK (VCN)

A private network that you set up in the Oracle data centers, with firewall rules and specific types of communication gateways that you can choose to use. A VCN covers a single, contiguous IPv4 CIDR block of your choice in the allowed IP address ranges.

You need at least one virtual cloud network before you launch a load balancer.

For information about setting up virtual cloud networks, see Overview of Networking.

VISIBILITY

Specifies whether your load balancer is public or private. A public load balancer has a public IP address that clients can access from the internet. A private load balancer has a private IP address, from a VCN local subnet, that clients can access only from within your VCN.

For more information, see Managing a Load Balancer.

WORK REQUEST

An object that reports on the current state of a Load Balancing request.

The Load Balancing service handles requests asynchronously. Each request returns a work request ID (OCID) as the response. You can view the work request item to see the status of the request.

For more information, see Viewing the State of a Work Request.

Resource Identifiers

Most types of Oracle Cloud Infrastructure resources have a unique, Oracle-assigned identifier called an Oracle Cloud ID (OCID). For information about the OCID format and other ways to identify your resources, see Resource Identifiers.

Ways to Access Oracle Cloud Infrastructure

You can access Oracle Cloud Infrastructure using the Console (a browser-based interface) or the REST API. Instructions for the Console and API are included in topics throughout this guide. For a list of available SDKs, see Software Development Kits and Command Line Interface.

To access the Console, you must use a supported browser. Oracle Cloud Infrastructure supports the latest desktop versions of Google Chrome, Microsoft Edge, Internet Explorer 11, Safari, Firefox, and Firefox ESR. Note that private browsing mode is not supported for Firefox, Internet Explorer, or Edge. Mobile browsers are not supported.

For general information about using the API, see REST APIs.

Monitoring Resources

You can monitor the health, capacity, and performance of your Oracle Cloud Infrastructure resources by using metrics, alarms, and notifications. For more information, see Monitoring Overview and Notifications Overview.

For information about monitoring the traffic passing through your load balancer, see Load Balancing Metrics.

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

Limits on Load Balancing Resources

See Service Limits for a list of applicable limits and instructions for requesting a limit increase.

Other limits include:

- You cannot dynamically change the load balancer shape to handle more incoming traffic. You can use the API or Console to create a load balancer with the new shape information.
- You cannot convert an AD-specific load balancer to a regional load balancer or the reverse.
- The Load Balancing service does not support IPv6 addresses.

- The maximum number of concurrent connections is limited when you use stateful security list rules for your load balancer subnets. In contrast, there is no theoretical limit on concurrent connections if you use stateless security list rules. The practical limitations depend on various factors. The larger your load balancer shape, the greater the connection capacity. Other considerations include system memory, TCP timeout periods, TCP connection state, and so forth.

  Tip

  To accommodate high-volume traffic, Oracle strongly recommends that you use stateless security list rules for your load balancer subnets.

- Each load balancer has the following configuration limits:
  - One IP address
  - 16 backend sets
  - 512 backend servers per backend set
  - 1024 backend servers total
  - 16 listeners

How Load Balancing Policies Work

After you create a load balancer, you can apply policies to control traffic distribution to your backend servers. The Load Balancing service supports three primary policy types:

- Round Robin
- Least Connections
- IP Hash

When processing load or capacity varies among backend servers, you can refine each of these policy types with backend server *weighting*. Weighting affects the proportion of requests directed to each server. For example, a server weighted '3' receives three times the number

of connections as a server weighted '1'. You assign weights based on criteria of your choosing, such as each server's traffic-handling capacity.

Load balancer policy decisions apply differently to TCP load balancers, cookie-based session persistent HTTP requests (sticky requests), and non-sticky HTTP requests.

- A TCP load balancer considers policy and weight criteria to direct an initial incoming request to a backend server. All subsequent packets on this connection go to the same endpoint.
- An HTTP load balancer configured to handle cookie-based session persistence forwards requests to the backend server specified by the cookie's session information.
- For non-sticky HTTP requests, the load balancer applies policy and weight criteria to every incoming request and determines an appropriate backend server. Multiple requests from the same client could be directed to different servers.

Round Robin

Round Robin is the default load balancer policy. This policy distributes incoming traffic sequentially to each server in a backend set list. After each server has received a connection, the load balancer repeats the list in the same order.

Round Robin is a simple load balancing algorithm. It works best when all the backend servers have similar capacity and the processing load required by each request does not vary significantly.

Least Connections

The Least Connections policy routes incoming non-sticky request traffic to the backend server with the fewest active connections. This policy helps you maintain an equal distribution of active connections with backend servers. As with the round robin policy, you can assign a weight to each backend server and further control traffic distribution.



Tip

In TCP use cases, a connection can be active but have no current traffic. Such connections do not serve as a good load metric.

IP Hash

The IP Hash policy uses an incoming request's source IP address as a hashing key to route non-sticky traffic to the same backend server. The load balancer routes requests from the same client to the same backend server as long as that server is available. This policy honors server weight settings when establishing the initial connection.

IP Hash ensures that requests from a particular client are always directed to the same backend server, as long as it is available.



Warning

Multiple clients that connect to the load balancer through a proxy or NAT router appear to have the same IP address. If you apply the IP Hash policy to your backend set, the load balancer routes traffic based on the incoming IP address and sends these proxied client requests to the same backend server. If the proxied client pool is large, the requests could flood a backend server.

Connection Management

Oracle Cloud Infrastructure load balancers support connection multiplexing. The load balancer can route many incoming requests from multiple clients to the destination backend server

through a few (one or multiple) backend connections.

After your load balancer connects a client to a backend server, the connection can be closed due to inactivity. Also, you can configure load balancer listeners to control the maximum idle time allowed during each TCP connection or HTTP request and response pair. Oracle recommends that you do not allow your backend servers to close connections to the load balancer.

Highlights

Three different timeout settings affect your load balancer's behavior:

- Keep-alive **setting between the load balancer and backend server**
  The load balancer closes backend server connections that are idle for more than 300 seconds.
- **Keep-alive setting between the load balancer and the client**
  The Load Balancing service sets the keep-alive value to maintain the connection for 10,000 transactions or until it has been idle for 65 seconds, whichever limit occurs first. You cannot change the value of this setting.
- Idle timeout
  You can set the duration of the idle timeout when you create a listener. This setting applies to the time allowed between two successive receive or two successive send network input/output operations during the HTTP request-response phase.

Keep-Alive Settings

The load balancing service does not honor keep-alive settings from backend servers. The load balancer closes backend server connections that are idle for more than 300 seconds. Oracle recommends that you do not allow your backend servers to close connections to the load balancer. To prevent possible 502 errors, ensure that your backend servers do not close idle connections in less than 310 seconds.

The Load Balancing service sets the keep-alive value to maintain the connection for 10,000 transactions or until it has been idle for 65 seconds, whichever limit occurs first. You cannot change the value of this setting.

Connection Configuration

When you create a TCP or HTTP listener, you can specify the maximum idle time in seconds. This setting applies to the time allowed between two successive receive or two successive send network input/output operations during the HTTP request-response phase. If the configured timeout has elapsed with no packets sent or received, the client's connection is closed. For HTTP and WebSocket connections, a send operation does not reset the timer for receive operations and a receive operation does not reset the timer for send operations.



Tip

This timeout setting does not apply to idle time between a completed response and a subsequent HTTP request.

The default timeout values are:

- 300 seconds for TCP listeners.
- 60 seconds for HTTP listeners.

Modify the timeout parameter if either the client or the backend server requires more time to transmit data. Some examples include:

- The client sends a database query to the backend server and the database takes over 300 seconds to execute. Therefore, the backend server does not transmit any data within 300 seconds.
- The client uploads data using the HTTP protocol. During the upload, the backend does not transmit any data to the client for more than 60 seconds.
- The client downloads data using the HTTP protocol. After the initial request, it stops transmitting data to the backend server for more than 60 seconds.

- The client starts transmitting data after establishing a WebSocket connection, but the backend server does not transmit data for more than 60 seconds.
- The backend server starts transmitting data after establishing a WebSocket connection, but the client does not transmit data for more than 60 seconds.

The maximum timeout value is 7200 seconds. Contact My Oracle Support to file a service request if you want to increase this limit for your tenancy. For more information, see Service Limits.

HTTP "X-" Headers

HTTP requests and responses often include header fields that provide contextual information about the message. RFC 2616 defines a standard set of HTTP header fields. Some non-standard header fields, which begin with `X-`, are common. The Load Balancing service adds or modifies the following `X-` headers when it passes requests to your servers.

X-Forwarded-For

Provides a list of connection IP addresses.

The load balancer appends the last remote peer address to the `X-Forwarded-For` field from the incoming request. A comma and space precede the appended address. If the client request header does not include an `X-Forwarded-For` field, this value is equal to the `X-Real-IP` value. The original requesting client is the first (left-most) IP address in the list, assuming that the incoming field content is trustworthy. The last address is the last (most recent) peer, that is, the machine from which the load balancer received the request. The format is:

```
X-Forwarded-For: <original_client>, <proxy1>, <proxy2>
```

Example incoming field:

```
X-Forwarded-For: 202.1.112.187
```

Example field with appended proxy IP address:

```
X-Forwarded-For: 202.1.112.187, 192.168.0.10
```

X-Forwarded-Host

Identifies the original host and port requested by the client in the `Host` HTTP request header. This header helps you determine the original host, since the hostname or port of the reverse proxy (load balancer) might differ from the original server handling the request.

```
X-Forwarded-Host: www.oracle.com:8080
```

X-Forwarded-Port

Identifies the listener port number that the client used to connect to the load balancer. For example:

```
X-Forwarded-Port: 443
```

X-Forwarded-Proto

Identifies the protocol that the client used to connect to the load balancer, either `http` or `https`. For example:

```
X-Forwarded-Proto: https
```

X-Real-IP

Identifies the client's IP address. For the Load Balancing service, the "client" is the last remote peer.

Your load balancer intercepts traffic between the client and your server. Your server's access logs, therefore, include only the load balancer's IP address. The `X-Real-IP` header provides the client's IP address. For example:

```
X-Real-IP: 192.168.0.10
```

Session Persistence

Session persistence is a method to direct all requests originating from a single logical client to a single backend web server. Backend servers that use caching to improve performance, or to

enable log-in sessions or shopping carts, can benefit from session persistence.

Your load balancer must operate in HTTP mode to support server side, cookie-driven session persistence. You can enable the session persistence feature when you create a backend set. To configure session persistence, you specify a cookie name and decide whether to disable fallback for unavailable servers. You can edit an existing backend set to change the session persistence configuration.

Cookies

The Load Balancing service activates session persistence when a backend server sends a `Set-Cookie` response header containing a recognized cookie name. The cookie name must match the name specified in the backend set configuration. If the configuration specifies a match-all pattern, '*', any cookie set by the server activates session persistence. Unless a backend server activates session persistence, the service follows the load balancing policy specified when you created the load balancer.

The client computer must accept cookies for Load Balancing session persistence feature to work.

The Load Balancing service calculates a hash of the configured cookie and other request parameters, and sends that value to the client in a cookie. The value stored in the cookie enables the service to route subsequent client requests to the correct backend server. If your backend servers change any of the defined cookies, the service recomputes the cookie's value and resends it to the client.



Warning

Oracle recommends that you treat cookie data as an opaque entity. Do not use it in your applications.

To stop session persistence, the backend server must delete the session persistence cookie. If you used the match-all pattern, it must delete all cookies. You can delete cookies by sending a `Set-Cookie` response header with a past expiration date. The Load Balancing service routes subsequent requests using the configured load balancing policy.



Note

IP Address-driven Session Persistence

Some products offer session persistence support without cookies. These products depend on the IP address of the incoming request. ISP proxies and company exit gateways can issue many requests from a single IP address. In this case, a single backend server can be subject to high traffic volumes. Your backend fleet can become overwhelmed, one server at a time, even though effective load balancing is possible.

Another weakness of IP address-driven session persistence is that the originating IP address can change. In this case, session persistence can be lost or the request redirected to the wrong backend server.

Fallback

By default, the Load Balancing service directs traffic from a persistent session client to a different backend server when the original server is unavailable. You can configure the backend set to disable this fallback behavior. When you disable fallback, the load balancer fails the request and returns an HTTP 502 code. The service continues to return an HTTP 502 until the client no longer presents a persistent session cookie.



Warning

If fallback is disabled, cookies with a distant future expiration date can cause a client outage.

The Load Balancing service considers a server marked `drain` available for existing persisted sessions. New requests that are not part of an existing persisted session are not sent to that server.

Managing a Load Balancer

This topic describes how to create or delete a load balancer on your system.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Prerequisites

To implement a working load balancer, you need:

- For a public load balancer in a region with multiple availability domains, you need a VCN with a public regional subnet or at least two public AD-specific subnets. In the latter case, each AD-specific subnet must reside in a separate availability domain. For more information on subnets, See VCNs and Subnets and Public vs. Private Subnets.



Warning

You cannot specify a private subnet for your public load balancer.

- For a public load balancer in a region with only one availability domain, you need a VCN with at least one public subnet.
- For a private load balancer in any region, you need a VCN with at least one private subnet.
- Two or more backend servers (Compute instances) running your applications. For more information on Compute instances, see Creating an Instance.



Note

Private IP Address Consumption

A public load balancer created in one public regional subnet consumes two private IP addresses from the host subnet. The primary and secondary load balancers reside within the same subnet. Each load balancer requires a private IP address from that subnet. The Load Balancing service assigns a floating public IP address, which does not come from the host subnet.

A public load balancer created in two public AD-specific subnets consumes two private IP addresses, one from each host subnet. The primary and secondary load balancers reside within different subnets. Each load balancer requires one private IP address from its host subnet. The Load Balancing service assigns a floating public IP address, which does not come from the host subnets.

A private load balancer created in a single subnet consumes three private IP addresses from the host subnet. The primary and secondary load balancers reside within the same subnet. Each load balancer requires a private IP address from that subnet. The floating private IP address also comes from the host subnet.

Working with Load Balancers

For background information on Oracle Cloud Infrastructure Load Balancing, see Overview of Load Balancing.

For the purposes of access control, you must specify the compartment where you want the load balancer to reside. Consult an administrator in your organization if you're not sure which compartment to use. For information about compartments and access control, see Overview of Oracle Cloud Infrastructure Identity and Access Management.

When you create a load balancer within your VCN, you get a public or private IP address, and provisioned total bandwidth. If you need another IP address, you can create another load balancer.

A public load balancer in a region with multiple availability domains requires one public regional subnet or two public AD-specific subnets to host the primary load balancer and a standby. In the latter case, each AD-specific subnet must reside in a separate availability domain. A public load balancer in a region with only one availability domain requires a single public subnet to host the primary load balancer and a standby. For more information on VCNs and subnets, see Overview of Networking. You can associate the public IPv4 address with a DNS name from any vendor. You can use the public IP address as a front end for incoming traffic. The load balancer can route data traffic to any backend server that is reachable from the VCN.

A private load balancer requires only one subnet to host the primary load balancer and a standby. The private IP address is local to the subnet. The load balancer is accessible only from within the VCN that contains the associated subnet, or as further restricted by your security list rules. The load balancer can route data traffic to any backend server that is reachable from the VCN.

The essential components for load balancing include:

- A load balancer with pre-provisioned bandwidth.
- A backend set with a health check policy. See Managing Backend Sets.
- Backend servers for your backend set. See Managing Backend Servers.
- One or more listeners. See Managing Load Balancer Listeners.
- Load balancer subnet security list rules to allow the intended traffic. To learn more about these rules, see Parts of a Security List Rule.



Tip

To accommodate high-volume traffic, Oracle strongly recommends that you use stateless security list rules for your load balancer subnets.

Optionally, you can associate your listeners with SSL server certificate bundles to manage how your system handles SSL traffic. See Managing SSL Certificates.

For information about the number of load balancers you can have, see Service Limits.

Configuration Changes and Service Disruption

For a running load balancer, some configuration changes lead to service disruptions. The following guidelines help you understand the effect of changes to your load balancer.

- Operations that add, remove, or modify a backend server create no disruptions to the Load Balancing service.
- Operations that edit an existing health check policy create no disruptions to the Load Balancing service.
- Operations that trigger a load balancer reconfiguration can produce a brief service disruption with the possibility of some terminated connections.

Health Status

The Load Balancing service provides health status indicators that use your health check policies to report on the general health of your load balancers and their components. You can see health status indicators on the Console *List* and *Details* pages for load balancers, backend sets, and backend servers. You also can use the Load Balancing API to retrieve this information.

For general information about health status indicators, see Editing Health Check Policies.

Load Balancer Health Summary

The Console list of load balancers provides health status summaries that indicate the overall health of each load balancer. There are four levels of health status indicators. The meaning of each level is:

- **OK:** All backend sets associated with the load balancer return a status of OK.
- **WARNING:** All the following conditions are true:
    - At least one backend set associated with the load balancer returns a status of WARNING or UNKNOWN.
    - No backend sets return a status of CRITICAL.
    - The load balancer life cycle state is ACTIVE.
- **CRITICAL:** At least one backend set associated with the load balancer returns a status of CRITICAL.
- **UNKNOWN:** Any one of the following conditions is true:
    - The load balancer life cycle state is not ACTIVE.
    - No backend sets are defined for the load balancer.
    - All the following conditions are true:
        - More than half of the backend sets associated with the load balancer return a status of UNKNOWN.
        - None of the backend sets return a status of WARNING or CRITICAL.
        - The load balancer life cycle state is ACTIVE.
    - The system could not retrieve metrics for any reason.

For guidance on detecting and correcting common issues, see Using Health Status.

Load Balancer Health Details

The load balancer *Details* page provides the same **Overall Health** status indicator found in the list of load balancers. It also includes counters for the **Backend Set Health** status values

reported by the load balancer's child backend sets.

The health status counter badges indicate the following:

- The number of child entities reporting the indicated health status level.
- If a counter corresponds to the overall health, the badge has a fill color.
- If a counter has a zero value, the badge has a light gray outline and no fill color.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: For a typical policy that gives access to load balancers and their components, see Let network admins manage load balancers.

Also, be aware that a policy statement with `inspect load-balancers` gives the specified group the ability to see *all* information about the load balancers. For more information, see Details for Load Balancing.

If you're new to policies, see Getting Started with Policies and Common Policies.

Using the Console

To create a load balancer

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
2. Choose a **Compartment** you have permission to work in, and then click **Create Load Balancer**.

3. In the **Create Load Balancer** dialog box, configure your load balancer.
   Load Balancer Information
   - **Name:** Required. Specify a friendly name for the load balancer. It does not have to be unique, but it cannot be changed in the Console. (You can, however, change it with the API.) Avoid entering confidential information.
   - **Shape:** Required. Specify a shape to provision the maximum total bandwidth (ingress plus egress) for your load balancer. Available shapes include:
     - 100 Mbps
     - 400 Mbps
     - 8000 Mbps

     Tip

     After you create a load balancer, you cannot change the shape. You can create another load balancer with the new shape.
   - **Visibility Type:** Specify whether your load balancer is public or private.
     - **Public Load Balancer:** Choose this option to create a public load balancer. You can use the assigned public IP address as a front end for incoming traffic and to balance that traffic across all backend servers.
     - **Private Load Balancer:** Choose this option to create a private load balancer. You can use the assigned private IP address as a front end for incoming internal VCN traffic and to balance that traffic across all backend servers.

   Network Information
   If the current compartment contains no virtual cloud networks, you must create one before you use this form. You must also create security list rules to allow traffic to the backend servers that you specify for your backend set.

By default, the Console shows a list of VCNs in the compartment you're currently working in. To select a VCN from a different compartment, use the **click here** link that appears near the top of the **Create Load Balancer** form. The following configuration fields appear:

- **Virtual Cloud Network Compartment:** Required, when this option appears. Specify the compartment from which to select your VCN resources.
- **Virtual Cloud Network:** Required. Specify a VCN for the load balancer.
- **Subnet Compartment:** Required, when this option appears. Specify the compartment from which to select your subnets.

  By default, the Console shows a list of subnets in the compartment you're currently working in. Use the **click here** link to select a subnet in a different compartment.
- **Subnet:** Required. Select an available subnet. For a public load balancer, it must be a public subnet.

  Tip

  Subnets can also be either AD-specific or regional (regional ones have "*regional*" after the name). Oracle recommends using regional subnets. For more information, see About Regional Subnets.
- **Subnet (2 of 2):** Required for a public load balancer, if you specified an AD-specific subnet for **Subnet (1 of 2)**. Select a second public subnet. The second subnet must reside in a separate availability domain from the first subnet.



Tip

If you chose to create a private load balancer under **Visibility Type**, the form prompts you to select only one subnet.

Listener Information

- **Protocol:** Required. Specify the protocol to use, either HTTP or TCP.
- **Port:** Required. Specify the port on which to listen for incoming traffic.
- **Use SSL:** Optional. Check this box to create an SSL certificate bundle and attach it to the listener.



Note

When you check this box, the **Port** field is set to "443".

The following settings are required to enable SSL handling. See Managing SSL Certificates for more information.

  - **Certificate Name:** Required. Specify a friendly name for the certificate bundle. It must be unique within the load balancer, and it cannot be changed in the Console. (It can be changed using the API.) Avoid entering confidential information.
  - **Certificate:** Optional. (Required for SSL termination.) Paste the certificate, in PEM format, in this field.
  - **CA Certificate:** Optional. (Recommended for backend SSL termination configurations.) Paste the Certificate Authority certificate (root CA certificate) in this field.

- **Private Key:** Optional. (Required for SSL termination.) Paste the private key for the certificate in this field.
- **Passphrase:** Optional. Specify a passphrase.

- **Advanced Options**
  - **Listener Name:** You can replace the default name with a friendly name for the listener. The name must be unique, and cannot be changed. Avoid entering confidential information.
  - **Idle Timeout in Seconds:** Optional. Specify the maximum idle time in seconds. This setting applies to the time allowed between two successive receive or two successive send network input/output operations during the HTTP request-response phase.



Tip

The maximum value is 7200 seconds. For more information, see Connection Management.

Backend Set Information

- **Traffic Distribution Policy:** Required. Choose the load balancer policy for the backend set. The available options are:
  - **IP Hash**
  - **Least Connections**
  - **Weighted Round Robin**

  For more information on these policies, see How Load Balancing Policies Work.
- **Backend Server IP Addresses:** The load balancer routes traffic to the specified IP addresses. The backend set stores the private IP address, port, and weight entries, but not the instance names.

- **Choose a Compute instance private IP address:** This option helps you find the right private IP address by Compute instance name.
  - **Compartment:** Required. Select the compartment that contains the Compute instance you want to add.
  - **Instance Name:** Required. Select a Compute instance from the drop-down list.
  - **Private IP Address:** This field shows the private IP address that corresponds to the instance name you selected.
  - **Port:** Required. Specify the server port to which the load balancer must direct traffic.
  - **Weight:** Optional. Specify the weight to apply to this server. For more information, see How Load Balancing Policies Work.
- **Enter a Compute instance private IP address:** This option adds a backend server to the backend set based on the Compute instance private IP address.
  - **Private IP Address:** Required. Specify the private IP address of the backend server (Compute instance) to add.
  - **Port:** Required. Specify the server port to which the load balancer must direct traffic.
  - **Weight:** Optional. Specify the weight to apply to this server.

- **Advanced options**
  - **Health Check:** Required. Specify the test parameters to confirm the health of backend servers.

- **Protocol:** Required. Specify the protocol to use, either HTTP or TCP.



Importa nt

Configure your health check protocol to match your application or service.

- **Port:** Required. Specify the backend server port against which to run the health check.



Tip

You can enter the value '0' to have the health check use the backend server's traffic port.

- **Interval in ms:** Optional. Specify how frequently to run the health check, in milliseconds. The default is 10000 (10 seconds).
- **Timeout in ms:** Optional. Specify the maximum time in milliseconds to wait for a reply to a health check. A health check is successful only if a reply returns within this timeout period. The default is 3000 (3 seconds).
- **Number of retries:** Optional. Specify the number of retries to attempt before a backend server is considered "unhealthy". The default is '3'.
- **URL Path (URI):** (HTTP only) Required. Specify a URL endpoint against which to run the health check.

        - **Status Code:** (HTTP only) Optional. Specify the status code a healthy backend server must return.
        - **Response Body Regex:** (HTTP only) Optional. Provide a regular expression for parsing the response body from the backend server.
    - **Backend Set:** Specify advanced backend set configuration.
        - **Backend Set Name:** Required. Specify a friendly name for the backend set. It must be unique within the load balancer, and it cannot be changed.

- **Use SSL:** Optional. Check this box to associate an SSL certificate bundle with the backend set. The following settings are required to enable SSL handling. See Managing SSL Certificates for more information.
  - **Certificate Name:** Required. Specify a friendly name for the certificate bundle. It must be unique within the load balancer, and it cannot be changed in the Console. (It can be changed using the API.) Avoid entering confidential information.
  - **Certificate:** Optional. (Required for SSL termination.) Paste the certificate, in PEM format, in this field.

    Important

    If you submit a self-signed certificate for backend SSL, you must submit the same certificate in the corresponding CA Certificate field.

  - **CA Certificate:** Optional. (Recommended for backend SSL termination configurations.) Paste the Certificate Authority certificate (root CA certificate) in this field.
  - **Private Key:** Optional. (Required for SSL termination.) Paste the private key for the certificate in this field.
  - **Passphrase:** Optional. Specify a passphrase.
  - **Verify Peer Certificate:** Optional. Select this option to enable peer certificate verification.

        - **Verify Depth:** Optional. Specify the maximum depth for certificate chain verification.
      - **Use Session Persistence:** Optional. Check this box to enable persistent sessions from a single logical client to a single backend web server. The following settings configure session persistence. See Session Persistence for more information.
        - **Cookie Name:** The cookie name used to enable session persistence. Specify '*' to match any cookie name.
        - **Disable Fallback:** Check this box to disable fallback when the original server is unavailable.
    - **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
    - **View detail page after this resource is created:** Check this box to jump to the **Load Balancer Details** page after your load balancer is created.
4. Click **Create**.

After the system provisions the load balancer, details appear in the load balancer list. To view more details, click the load balancer name.

To delete a load balancer

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
2. Choose the **Compartment** that contains the load balancer you want to delete.
3. For the load balancer you want to delete, click the Actions icon (three dots), and then click **Delete**.

4. Confirm when prompted.

Managing Tags for a Load Balancer

You can apply tags to your resources, such as load balancers, to help you organize them according to your business needs. You can apply tags at the time you create a load balancer, or you can update the load balancer later with the desired tags. For general information about applying tags, see Resource Tags.

To manage tags for a load balancer

1. Open the Console, click **Networking**, and then click **Load Balancers**.
2. Choose the **Compartment** that contains the load balancer you want to tag, and then click the load balancer's name.
3. Click the **Tags** tab to view or edit existing tags, or click **Apply Tag(s)** to add new ones.

For more information, see Resource Tags.

Monitoring Resources

You can monitor the health, capacity, and performance of your Oracle Cloud Infrastructure resources by using metrics, alarms, and notifications. For more information, see Monitoring Overview and Notifications Overview.

For information about monitoring the traffic passing through your load balancer, see Load Balancing Metrics.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage load balancers:

- CreateLoadBalancer
- DeleteLoadBalancer
- GetLoadBalancer
- GetLoadBalancerHealth
- ListLoadBalancers
- ListLoadBalancerHealths
- UpdateLoadBalancer: You can update the load balancer's display name.

Managing Backend Sets

This topic describes how to create and delete backend sets for use with a load balancer. For information about managing load balancers, see Managing a Load Balancer.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: For a typical policy that gives access to load balancers and their components, see Let network admins manage load balancers.

Also, be aware that a policy statement with `inspect load-balancers` gives the specified group the ability to see *all* information about the load balancers. For more information, see Details for Load Balancing.

If you're new to policies, see Getting Started with Policies and Common Policies.

Working with Backend Sets

A backend set is a logical entity defined by a load balancing policy, a health check policy, and a list of backend servers. To create a backend set, you must specify a load balancing policy and health check script, and then add a list of backend servers (Compute instances). SSL and session persistence configuration is optional. A backend set must be associated with one or more listeners for the load balancer to work.

You cannot delete a backend set used by an active listener.

Changing the load balancing policy of a backend set temporarily interrupts traffic and can drop active connections.

For background information on the Oracle Cloud Infrastructure Load Balancing, see Overview of Load Balancing.

Health Status

The Load Balancing service provides health status indicators that use your health check policies to report on the general health of your load balancers and their components. You can see health status indicators on the Console *List* and *Details* pages for load balancers, backend sets, and backend servers. You also can use the Load Balancing API to retrieve this information.

For general information about health status indicators, see Editing Health Check Policies.

Backend Set Health Summary

The Console list of a load balancer's backend sets provides health status summaries that indicate the overall health of each backend set. There are four levels of health status

indicators. The meaning of each level is:

- **OK:** All backend servers in the backend set return a status of OK.
- **WARNING:** Both of the following conditions are true:
  - Half or more of the backend set's backend servers return a status of OK.
  - At least one backend server returns a status of WARNING, CRITICAL, or UNKNOWN.
- **CRITICAL:** Fewer than half of the backend set's backend servers return a status of OK.
- **UNKNOWN:** At least one of the following conditions is true:
  - More than half of the backend set's backend servers return a status of UNKNOWN.
  - The system could not retrieve metrics for any reason.
  - The backend set does not have a listener attached.

For guidance on detecting and correcting common issues, see Using Health Status.

Backend Set Health Details

The backend set *Details* page provides the same **Overall Health** status indicator found in the load balancer's list of backend sets. It also includes counters for the **Backend Health** status values reported by the backend set's child backend servers.

The health status counter badges indicate the following:

- The number of child entities reporting the indicated health status level.
- If a counter corresponds to the overall health, the badge has a fill color.
- If a counter has a zero value, the badge has a light gray outline and no fill color.

Using the Console

To create a backend set

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
2. Click the name of the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
3. In the **Resources** menu, click **Backend Sets** (if necessary), and then click **Create Backend Set**.
4. In the **Create Backend Set** dialog box, enter the following:
   - **Name:** Required. Specify a friendly name for the backend set. It must be unique within the load balancer, and it cannot be changed.
     Valid backend set names include only alphanumeric characters, dashes, and underscores. Backend set names cannot contain spaces. Avoid entering confidential information.
   - **Policy:** Required. Choose the load balancer policy for the backend set. The available options are:
     - **IP Hash**
     - **Least Connections**
     - **Weighted Round Robin**

     For more information on these policies, see How Load Balancing Policies Work.

- **Use SSL:** Optional. Check this box to associate an SSL certificate bundle with the backend set. The following settings are required to enable SSL handling. See Managing SSL Certificates for more information.
    - **Certificate Name:** Required. The friendly name of the SSL certificate to use. See Managing SSL Certificates for more information.
    - **Verify Peer Certificate:** Optional. Select this option to enable peer certificate verification.
    - **Verify Depth:** Optional. Specify the maximum depth for certificate chain verification.
- **Use Session Persistence:** Optional. Check this box to enable persistent sessions from a single logical client to a single backend web server. The following settings configure session persistence. See Session Persistence for more information.
    - **Cookie Name:** The cookie name used to enable session persistence. Specify '*' to match any cookie name.
    - **Disable Fallback:** Check this box to disable fallback when the original server is unavailable.
- **Health Check:** Required. Specify the test parameters to confirm the health of backend servers.
    - **Protocol:** Required. Specify the protocol to use, either HTTP or TCP.



Important

Configure your health check protocol to match your application or service.

- **Port:** Required. Specify the backend server port against which to run the health check.

  Tip

  You can enter the value '0' to have the health check use the backend server's traffic port.

- **Interval in ms:** Optional. Specify how frequently to run the health check, in milliseconds. The default is 10000 (10 seconds).
- **Timeout in ms:** Optional. Specify the maximum time in milliseconds to wait for a reply to a health check. A health check is successful only if a reply returns within this timeout period. The default is 3000 (3 seconds).
- **Number of retries:** Optional. Specify the number of retries to attempt before a backend server is considered "unhealthy". The default is '3'.
- **URL Path (URI):** (HTTP only) Required. Specify a URL endpoint against which to run the health check.
- **Status Code:** (HTTP only) Optional. Specify the status code a healthy backend server must return.
- **Response Body Regex:** (HTTP only) Optional. Provide a regular expression for parsing the response body from the backend server.

5. Click **Create**.

After your backend set is provisioned, you must specify backend servers for the set. See Managing Backend Servers for more information.

To edit a backend set



Warning

Updating the backend set temporarily interrupts traffic and can drop active connections.

When you edit a backed set, you can choose a new load balancing policy and modify the SSL configuration.

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
2. Click the name of the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
3. In the **Resources** menu, click **Backend Sets** (if necessary).
4. Click the name of the backend set you want to edit.
5. Click **Edit Backend Set**.
6. Make the configuration changes you need, and then click **Submit**.

If you want to modify the backend set's health check policy, see Editing Health Check Policies.

If you want to add or remove backend servers from the backend set, see Managing Backend Servers.

To delete a backend set



Tip

You cannot delete a backend set used by an active listener. First, remove any backend sets you want to



delete from the associated listeners.

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
2. Click the name of the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
3. In the **Resources** menu, click **Backend Sets** (if necessary).
4. For the backend set you want to delete, click the Actions icon (three dots), and then click **Delete**.
5. Confirm when prompted.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage load balancer backend sets:

- CreateBackendSet
- DeleteBackendSet
- GetBackendSet
- GetBackendSetHealth
- ListBackendSets
- UpdateBackendSet

Managing Backend Servers

This topic describes how to manage backend servers for use with a load balancer. For information about managing load balancers, see Managing a Load Balancer.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: For a typical policy that gives access to load balancers and their components, see Let network admins manage load balancers.

Also, be aware that a policy statement with `inspect load-balancers` gives the specified group the ability to see *all* information about the load balancers. For more information, see Details for Load Balancing.

If you're new to policies, see Getting Started with Policies and Common Policies.

Working with Backend Servers

When you implement a load balancer, you must specify the backend servers (Compute instances) to include in each backend set. The load balancer routes incoming traffic to these backend servers based on the policies you specified for the backend set. You can use the Console to add and remove backend servers in a backend set.

To route traffic to a backend server, the Load Balancing service requires the IP address of the compute instance and the relevant application port. If the backend server resides within the same VCN as the load balancer, Oracle recommends that you specify the compute instance's private IP address. If the backend server resides within a different VCN, you must specify the public IP address of the compute instance. You also must ensure that the VCN's security list rules allow Internet traffic.



Warning

When you add backend servers to a backend set, you specify either the instance OCID or an IP address for the server to add. An instance with multiple VNICs attached can have multiple IP addresses pointing to it.

- If you identify a backend server by OCID, Load Balancing uses the primary VNIC's primary private IP address.
- If you identify the backend servers to add to a backend set by their IP addresses, it is possible to point to the same instance more than once.

To enable backend traffic, your backend server subnets must have appropriate ingress and egress rules in their security lists. When you add backend servers to a backend set, the Load Balancing service Console can suggest security list rules for you. If you do not allow the service to create security list rules, you must create your own rules using the Networking service. To learn more about these rules, see Parts of a Security List Rule.



Tip

To accommodate high-volume traffic, Oracle strongly recommends that you use stateless security list rules for your load balancer subnets.

You can add and remove backend servers without disrupting traffic.

Health Status

The Load Balancing service provides health status indicators that use your health check policies to report on the general health of your load balancers and their components. You can see health status indicators on the Console *List* and *Details* pages for load balancers, backend sets, and backend servers. You also can use the Load Balancing API to retrieve this information.

For general information about health status indicators, see Editing Health Check Policies.

Backend Server Health Summary

The Console list of a backend set's backend servers provides health status summaries that indicate the overall health of each backend server. The primary and standby load balancers both provide health check results that contribute to the health status. There are four levels of health status indicators. The meaning of each level is:

- **OK:** The primary and standby load balancer health checks both return a status of OK.
- **WARNING:** One health check returned a status of OK and one did not.
- **CRITICAL:** Neither health check returned a status of OK.
- **UNKNOWN:** One or both health checks returned a status of UNKNOWN or the system was unable to retrieve metrics.

To view the health status details for a specific backend server, click its **IP Address**.

For guidance on detecting and correcting common issues, see Using Health Status.

Backend Server Health Details

The *Details* page for a backend set provides the same **Overall Health** status indicator found in the backend set's list of backend servers. It also reports the following data for the two health checks performed against each backend server:

IP ADDRESS

The IP address of the health check status report provider, which is a Compute instance managed by the Load Balancing service. This identifier helps you differentiate same-subnet load balancers that report health check status.

The Load Balancing service ensures high availability by providing one primary and one standby load balancer. To diagnose a backend server issue, you must know the source of the health check report. For example, a misconfigured security list might cause one load balancer instance to report that a backend server is healthy. The other load balancer instance might return an unhealthy status. In this case, one of the two load balancer instances cannot communicate with the backend server. Reconfigure the security list to restore the backend server's health status.

STATUS

The status returned by the health check. Possible values include:

- OK
  The backend server's response satisfied the health check policy requirements.
- INVALID_STATUS_CODE
  The HTTP response status code did not match the expected status code specified by the health policy.
- TIMED_OUT
  The backend server did not respond within the timeout interval specified by the health policy.
- REGEX_MISMATCH
  The backend server response did not satisfy the regular expression specified by the health policy.
- CONNECT_FAILED
  The health check server could not connect to the backend server.

- IO_ERROR
  An input or output communication error occurred while reading or writing a response or request to the backend server.
- OFFLINE
  The backend server is set to **offline**, so health checks are not run.
- UNKNOWN
  Health check status is not available.

LAST CHECKED

The date and time of the most recent health check.

Health status is updated every three minutes. No finer granularity is available.

Using the Console

To add one or more servers to a backend set

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
2. Click the name of the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
3. In the **Resources** menu, click **Backend Sets** if necessary. Click the name of the backend set to which you want to add one or more backend servers.

   Tip

   If the load balancer has no backend sets, you must create one before you can specify a backend server.

4. Click **Edit Backends**.

5. In the **Edit Backends** dialog box, enter the following:
   - **Help me create proper security list rules:** Select this check box if you want the Load Balancing service to suggest basic security list rules to enable traffic for this backend server.
   - **Instance OCID / IP Address:** Required. Specify either the OCID or IP address of the backend server (Compute instance) to add.

     If you chose to have the service help you create security list rules, you must specify the OCID. If you do not know the instance OCID, click **View Instances** to open the **Instances** page of the Compute service. There you can find the instance you want and copy its OCID.
   - **Port:** Required. Specify the server port to which the load balancer must direct traffic.
   - **Weight:** Optional. Specify the weight to apply to this server. For more information, see How Load Balancing Policies Work.
6. Click **Submit**. If you did not choose to have the service create security list rules for you, the specified servers are added.

   If you chose to have the service create security list rules for you, continue with the next step.
7. Review the suggested rules to be added to the security list rules for the indicated subnets. To add the rules, click **Add All Security Rules**.

To remove a server from a backend set

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
2. Click the name of the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
3. In the **Resources** menu, click **Backend Sets** if necessary. Click the name of the backend set from which you want to remove a backend server.
4. Click **Edit Backends**.

5. In the **Edit Backends** dialog box, click the red box to remove the corresponding server from the backend set.
6. Click **Submit**.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

The API enables you to mark a backend server in the following ways:

BACKUP

The load balancer forwards ingress traffic to this backend server only when all other backend servers not marked as "backup" fail the health check policy. This configuration is useful for handling disaster recovery scenarios.

DRAIN

The load balancer stops forwarding new TCP connections and new non-sticky HTTP requests to this backend server so an administrator can take the server out of rotation for maintenance purposes.

OFFLINE

The load balance forwards no ingress traffic to this backend server.

You also can use the API to specify a server's load balancing policy weight. For more information on load balancing policies, see How Load Balancing Policies Work.

Use these API operations to manage the backend servers in a backend set:

- CreateBackend
- DeleteBackend
- GetBackend

- GetBackendHealth
- ListBackends
- UpdateBackend

Managing Load Balancer Listeners

This topic is part of the setup and maintenance of a load balancer. For more Load Balancing information about managing load balancers, see Managing a Load Balancer.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: For a typical policy that gives access to load balancers and their components, see Let network admins manage load balancers.

Also, be aware that a policy statement with `inspect load-balancers` gives the specified group the ability to see *all* information about the load balancers. For more information, see Details for Load Balancing.

If you're new to policies, see Getting Started with Policies and Common Policies.

Working with Listeners

A listener is a logical entity that checks for incoming traffic on the load balancer's IP address.

To handle TCP, HTTP, and HTTPS traffic, you must configure at least one listener per traffic type.

When you create a listener, you must ensure that your VCN's security list rules allow the listener to accept traffic.



Tip

To accommodate high-volume traffic, Oracle strongly recommends that you use stateless security list rules for your load balancer subnets.

You can have one SSL certificate bundle per listener. You can configure two listeners, one each for ports 443 and 8443, and associate SSL certificate bundles with each listener. For more information about SSL certificates for load balancers, see Managing SSL Certificates.

Using the Console

To create a listener

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
2. Choose the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
3. In the **Resources** menu, click **Listeners** (if necessary), and then click **Create Listener**.
4. In the **Create Listener** dialog box, enter the following:

- **Name:** Required. Specify a friendly name for the listener. The name must be unique, and cannot be changed. Avoid entering confidential information.
- **Hostname:** Optional. Select up to 16 virtual hostnames for this listener.

> **Important**
>
> To apply a virtual hostname to a listener, the name must be part of the load balancer's configuration. If the load balancer has no associated hostnames, you can create one on the **Hostnames** page.

- **Protocol:** Required. Specify the protocol to use, either HTTP or TCP.
- **Port:** Required. Specify the port on which to listen for incoming traffic.
- **Use SSL:** Optional. Check this box to associate an SSL certificate bundle with the listener. The following settings are required to enable SSL handling. See Managing SSL Certificates for more information.
  - **Certificate Name:** Required. The friendly name of the SSL certificate bundle to use.
  - **Verify Peer Certificate:** Optional. Select this option to enable peer certificate verification.
  - **Verify Depth:** Optional. Specify the maximum depth for certificate chain verification.
- **Backend Set:** Required. Specify the default backend set to which the listener routes traffic.
- **Timeout in seconds:** Optional. Specify the maximum idle time in seconds. This setting applies to the time allowed between two successive receive or two successive send network input/output operations during the HTTP request-response phase.



Tip

The maximum value is 7200 seconds. For more information, see Connection Management.

- **Path Route Set:** Optional. Specify the name of the set of path-based routing rules that applies to this listener's traffic.



Important

- To apply a path route set to a listener, the set must be part of the load balancer's configuration.
- To remove a path route set from an existing listener, choose **None** as the **Path Route Set** option. The path route set remains available for use by other listeners on this load balancer.

- **Rule Sets:** Optional. Select a rule set to apply to this listener's traffic.



Important

- To apply a rule set to a listener, the set must be part of the load balancer's configuration.
- To remove a rule set from the list, click the corresponding red box. The rule set remains available for use by other listeners on this load balancer.

5. Click **Create**.

When you create a listener, you must also update your VCN's security list rules to allow traffic to that listener.

To edit a listener

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
2. Choose the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
3. In the **Resources** menu, click **Listeners** (if necessary).
4. For the listener you want to edit, click the Actions icon (three dots), and then click **Edit Listener**.
5. Make the configuration changes you need, and then click **Submit**.

To delete a listener

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.

2. Choose the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
3. In the **Resources** menu, click **Listeners** (if necessary).
4. For the listener you want to delete, click **Delete**.
5. Confirm when prompted.

To enable a listener to accept traffic

To enable a listener to accept traffic, you must update your VCN's security list rules:

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
   The list of VCNs in the current compartment appears.
2. Click the name of the VCN containing your load balancer, and then click **Security Lists**.
   A list of the security lists in the cloud network appears.
3. Click the name of the security list that applies to your load balancer.
4. Click **Edit All Rules**.
5. Under **Allow Rules for Ingress**, click **Add Rule**.
6. Configure the ingress rule to allow access from known resources only.
   For details on ingress rule configuration, see Parts of a Security List Rule.
7. Click **Save Security List Rules**.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage listeners:

- CreateListener
- DeleteListener
- UpdateListener

Managing Request Routing

This topic describes how to manage your load balancer's request routing. For information about managing load balancers, see Managing a Load Balancer.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: For a typical policy that gives access to load balancers and their components, see Let network admins manage load balancers.

Also, be aware that a policy statement with `inspect load-balancers` gives the specified group the ability to see *all* information about the load balancers. For more information, see Details for Load Balancing.

If you're new to policies, see Getting Started with Policies and Common Policies.

Routing Incoming Requests

The Load Balancing service enables you to route incoming requests to various backend sets. You can:

- Assign virtual hostnames to a listener.
- Create path route rules.
- Combine these techniques.

Virtual Hostnames

You can assign virtual hostnames to any listener you create for your load balancer. Each hostname can correspond to an application served from your backend. Some advantages of virtual hostnames include:

- A single associated IP address. Multiple hostnames, backed by DNS entries, can point to the same load balancer IP address.
- A single load balancer. You do not need a separate load balancer for each application.
- A single load balancer shape. Running multiple applications behind a single load balancer helps you manage aggregate bandwidth demands and optimize utilization.
- Simpler backend set management. Managing a set of backend servers under a single resource simplifies network configuration and administration.

You can define exact virtual hostnames, such as "app.example.com", or you can use wildcard names. Wildcard names include an asterisk (*) in place of the first or last part of the name. When searching for a virtual hostname, the service chooses the first matching variant in the following priority order:

1. Exact name match (no asterisk), such as `app.example.com`.
2. Longest wildcard name that begins with an asterisk, such as `*.example.com`.



Tip

Prefix wildcard names might require a wildcard certificate for HTTPS sites.

3. Longest wildcard name that ends with an asterisk, such as `app.example.*`.



Tip

Suffix wildcard names might require a multi-domain Subject Alternative Name (SAN) certificate for HTTPS sites.

You do not need to specify the matching pattern to apply. The pattern is inherent in the asterisk position, that is, starting, ending, or none.

The following considerations apply to virtual hostnames:

- You cannot use regular expressions.
- To apply virtual hostnames to a listener, you first create one or more virtual hostnames associated with a load balancer.
- Virtual hostname selection priority is not related to the listener's configuration order.
- You can apply a maximum of 16 virtual hostnames to a listener.
- You can associate a maximum of 16 virtual hostnames with a load balancer.



Tip

The virtual hostnames feature supports HTTP and HTTPS listeners only, but does not support TCP listeners.



Note

Default Listener

If a listener has no virtual hostname specified, that listener is the default for the assigned port.

If all listeners on a port have virtual hostnames, the first virtual hostname configured for that port serves as the default listener.

Path Route Rules

Some applications have multiple endpoints or content types, each distinguished by a unique URI path. For example, `/admin/`, `/data/`, `/video/`, or `/cgi/`. You can use path route rules to route traffic to the correct backend set without using multiple listeners or load balancers.

A *path route* is a string that the Load Balancing service matches against an incoming URI to determine the appropriate destination backend set.

- You cannot use asterisks in path route strings.
- You cannot use regular expressions.
- Path route string matching is case-insensitive.



Important

Browsers often add an ending slash to the path in a request. If you specify a path such as `/admin`, you might want to configure the path both with and without the trailing slash. For example,`/admin` and `/admin/`.

A *path route rule* consists of a path route string and a pattern match type.

- Pattern match types include:
  - **EXACT_MATCH**
    Looks for a path string that exactly matches the incoming URI path.
    Applies case-insensitive regex:

    ```
    ^<path_string>$
    ```

  - **FORCE_LONGEST_PREFIX_MATCH**
    Looks for the path string with the best, longest match of the beginning portion of the incoming URI path.
    Applies case-insensitive regex:

    ```
    <path_string>.*
    ```

  - **PREFIX_MATCH**
    Looks for a path string that matches the beginning portion of the incoming URI path.
    Applies case-insensitive regex:

    ```
    ^<path_string>.*
    ```

  - **SUFFIX_MATCH**
    Looks for a path string that matches the ending portion of the incoming URI path.
    Applies case-insensitive regex:

    ```
    .*<path_string>$
    ```

- Path route rules apply only to HTTP and HTTPS requests and have no effect on TCP requests.

A *path route set* includes all path route rules that define the data routing for a particular listener.

- You can specify up to 20 path route rules per path route set.
- You can have one path route set per listener. The maximum number of listeners limits the number of path route sets you can specify for a load balancer.

RULE PRIORITY

The system applies the following priorities, based on match type, to the path route rules within a set:

- For one path route rule that specifies the EXACT_MATCH type, there is no cascade of priorities. The listener looks for an exact match only.
- For two path route rules, one that specifies the EXACT_MATCH type and one that specifies any other match type, the exact match rule is evaluated first. If no match is found, then the system looks for the second match type.
- For multiple path route rules specifying various match types, the system applies the following priority cascade:
    1. EXACT_MATCH
    2. FORCE_LONGEST_PREFIX_MATCH
    3. PREFIX_MATCH or SUFFIX_MATCH
- The order of the rules within the path route set does not matter for EXACT_MATCH and FORCE_LONGEST_PREFIX_MATCH. The system applies the priority cascade no matter where these match types appear in the path route set.
- If matching cascades down to prefix or suffix matching, the order of the rules within the path route set DOES matter. The system chooses the first prefix or suffix rule that matches the incoming URI path.

Virtual Hostname and Path Route Rules Combinations

Virtual hostnames and path route rules route requests to backend sets. Listeners with a virtual hostname receive priority over the default (no hostname) listener. The following example shows the results of a simple routing interaction.

The example system includes three listeners and one path route set:

Listener 1

- Virtual hostname: *none*
- Default backend set: `A`

- Path route set: `PathRouteSet1`

Listener 2

- Virtual hostname: `captive.com`
- Default backend set: `B`
- Path route set: `PathRouteSet1`

Listener 3

- Virtual hostname: `wild.com`
- Default backend set: `C`
- Path route set: `PathRouteSet1`

Path Route Set

- Path route set name: `PathRouteSet1`
  - Exact match on path string `/tame/` routes to backend set `B`.
  - Exact match on path string `/feral/` routes to backend set `C`.

The example configuration routes incoming URLs as follows:

`http://animals.com/` is routed to backend set `A`

- Virtual hostname `animals.com` matches **Listener 1**.
- Path `/` is not an EXACT_MATCH for any path route string in `PathRouteSet1`.

`http://animals.com/tame/` is routed to backend set `B`

- Virtual hostname `animals.com` matches **Listener 1**.
- Path `/tame/` is an EXACT_MATCH for path route string `/tame/` in `PathRouteSet1`.

`http://animals.com/feral/` is routed to backend set `C`

- Virtual hostname `animals.com` matches **Listener 1**.
- Path `/feral/` is an EXACT_MATCH for path route string `/feral/` in `PathRouteSet1`.

`http://captive.com/` is routed to backend set `B`

- Virtual hostname `captive.com` matches **Listener 2**.
- Path `/` is not an EXACT_MATCH for any path route string in `PathRouteSet1`.

`http://captive.com/tame/` is routed to backend set `B`

- Virtual hostname `captive.com` matches **Listener 2**.
- Path `/tame/` is an EXACT_MATCH for path route string `/tame/` in `PathRouteSet1`.

`http://captive.com/feral/` is routed to backend set `C`

- Virtual hostname `captive.com` matches **Listener 2**.
- Path `/feral/` is an EXACT_MATCH for path route string `/feral/` in `PathRouteSet1`.

`http://wild.com/` is routed to backend set `C`

- Virtual hostname `wild.com` matches **Listener 3**.
- Path `/` is not an EXACT_MATCH for any path route string in `PathRouteSet1`.

`http://wild.com/tame/` is routed to backend set `B`

- Virtual hostname `wild.com` matches **Listener 3**.
- Path `/tame/` is an EXACT_MATCH for path route string `/tame/` in `PathRouteSet1`.

`http://wild.com/tame/` is routed to backend set `C`

- Virtual hostname `wild.com` matches **Listener 3**.
- Path `/feral/` is an EXACT_MATCH for path route string `/feral/` in `PathRouteSet1`.

Using the Console

You can specify virtual hostnames and path route sets when you create or update a listener.

Creating Virtual Hostnames

To apply virtual hostnames to a listener, you first create one or more virtual hostnames. The virtual hostnames become a part of the load balancer's configuration. You then specify one or more virtual hostnames to use when you create or update a listener for the load balancer.

To create a virtual hostname

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
2. Choose the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
3. In the **Resources** menu, click **Hostnames** (if necessary), and then click **Create Hostname**.
4. In the **Create Hostname** dialog box, enter the following:
    - **Name:** Required. Specify a friendly name for the hostname. The name must be unique, and cannot be changed. Avoid entering confidential information.
    - **Hostname:** Required. Specify the virtual hostname. See Virtual Hostnames for a description of valid hostname construction and behavior.
5. Click **Create**. The **Work Request Submitted** dialog box opens.

6. To close the dialog box, click **Close**. To open the **Work Requests** page and view the status of the work request, click **View All Work Requests**.

After you create a virtual hostname, the name becomes available for use with the associated load balance. To apply the hostname, create or update a listener.

To update a virtual hostname

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
2. Choose the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
3. In the **Resources** menu, click **Hostnames** (if necessary).
4. For the hostname you want to edit, click the Actions icon (three dots), and then click **Edit**.
5. In the **Edit Hostname** dialog box, enter your updates to the **Hostname** field.
   You cannot edit the **Name** field of an existing virtual hostname.
6. Click **Update**. The **Work Request Submitted** dialog box opens.
7. To close the dialog box, click **Close**. To open the **Work Requests** page and view the status of the work request, click **View All Work Requests**.

To delete a virtual hostname

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
2. Choose the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
3. In the **Resources** menu, click **Hostnames** (if necessary).

4. For the hostname you want to edit, click the Actions icon (three dots), and then click **Delete**. The **Work Request Submitted** dialog box opens.
5. To close the dialog box, click **Close**. To open the **Work Requests** page and view the status of the work request, click **View All Work Requests**.

Creating Path Route Sets

To apply path route rules to a listener, you first create a path route set that contains the rules. The path route set becomes a part of the load balancer's configuration. You then specify the path route set to use when you create or update a listener for the load balancer. To remove a path route set from a listener, edit the listener and choose **None** as the **Path Route Set** option.

To create a path route set

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
2. Choose the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
3. In the **Resources** menu, click **Path Route Sets** (if necessary), and then click **Create Path Route Set**.
4. In the **Create Path Route Set** dialog box, enter the following:
    - **Name:** Required. Specify a friendly name for the path route set. The name must be unique, and cannot be changed. Avoid entering confidential information.
    - **Path route rules**
        - **Order:** Optional. If you have multiple path route rules, you can click the up or down arrows to move the corresponding rule.



Tip

The order of the rules within the path route set does not matter in most cases. However, if matching cascades down to prefix or suffix matching, the system chooses the first prefix or suffix rule that matches the incoming URI path.

- **Match Type:** Required. The type of matching to apply to incoming URIs.
- **URL String:** Required. The path string to match against the incoming URI path, for example `/admin/`.
- **Backend Set Name:** Required. The name of the target backend set for requests where the incoming URI matches the specified path.

5. (Optional) Click **Add Line** to create another path route rule or click the red box to delete an existing rule. You can have up to 20 path route rules in a set.
6. Click **Create**.

After you create a path route set, the set becomes available for use with the associated load balance. Create or update a listener to apply the path route set.

To update a path route set

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
2. Choose the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
3. In the **Resources** menu, click **Path Route Sets** (if necessary).
4. Click the name of the path route set you want to update, and then click **Edit Path Route Rules**.

5. In the **Edit Path Route Rules** dialog box, edit the following as needed for each rule you want to change:
    - **Order:** Optional. If you have multiple path route rules, you can click the up or down arrows to move the corresponding rule.

      Tip

      The order of the rules within the path route set does not matter in most cases. However, if matching cascades down to prefix or suffix matching, the system chooses the first prefix or suffix rule that matches the incoming URI path.

    - **Match Type:** Required. The type of matching to apply to incoming URIs.
    - **URL String:** Required. The path string to match against the incoming URI path, for example `/admin/`.
    - **Backend Set Name:** Required. The name of the target backend set for requests where the incoming URI matches the specified path.
6. (Optional) Click **Add Line** to create another path route rule or click the red box to delete an existing rule. You can have up to 20 path route rules in a set.
7. Click **Save**.

To update a single path route rule

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
2. Choose the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
3. In the **Resources** menu, click **Path Route Sets** (if necessary).

4. Click the name of the path route set you want to update.
5. For the path route rule you want to edit, click the Actions icon (three dots), and then click **Edit Path Route**.
6. In the **Edit Path Route Rule** dialog box, edit the following as needed for each rule you want to change:
    - **Order:** Optional. If you have multiple path route rules, you can click the up or down arrows to move the corresponding rule.

      Tip

      The order of the rules within the path route set does not matter in most cases. However, if matching cascades down to prefix or suffix matching, the system chooses the first prefix or suffix rule that matches the incoming URI path.

    - **Match Type:** Required. The type of matching to apply to incoming URIs.
    - **URL String:** Required. The path string to match against the incoming URI path, for example `/admin/`.
    - **Backend Set Name:** Required. The name of the target backend set for requests where the incoming URI matches the specified path.
7. Click **Save**.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage request routing:

- CreateListener
- CreatePathRouteSet
- DeleteListener
- DeletePathRouteSet
- GetPathRouteSet
- ListPathRouteSets
- UpdateListener
- UpdatePathRouteSet

Managing Rule Sets

This topic describes how you can create rule sets composed of actions to apply to requests or responses at the listener. For example, you can add, alter, or remove HTTP headers. For more information about managing load balancer listeners, see Managing Load Balancer Listeners.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have

permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: For a typical policy that gives access to load balancers and their components, see Let network admins manage load balancers.

Also, be aware that a policy statement with `inspect load-balancers` gives the specified group the ability to see *all* information about the load balancers. For more information, see Details for Load Balancing.

If you're new to policies, see Getting Started with Policies and Common Policies.

Working with Rule Sets

A rule set is a named set of rules associated with a load balancer and applied to one or more listeners on that load balancer. Rules are objects that represent actions applied to requests or responses at a load balancer listener. Possible actions include adding, altering, or removing HTTP headers. You can associate a maximum of 50 rules with a load balancer.

Rule sets are not shared between load balancers. To use the same set of rules on an additional load balancer, you must create a new, identical rule set under that load balancer.

You can apply an existing rule set when you create or edit a listener. You can apply the same rule set to multiple listeners on the same load balancer.

Rules sets can help you to pass metadata to your backend servers to do things like:

- Identify which listener sent a request.
- Notify a backend server about SSL termination.

Examples of how rule sets can help you enhance site security include:

- Adding headers to prevent external domains from iframing your site.
- Removing debug headers, such as "Server", sent by backend servers. This action helps you hide the implementation details of your backend.
- Adding the "strict-transport-security" header, with a proper value, to responses. This header helps guarantee that access to your site is HTTPS only.

- Adding the "x-xss-protection" header with a proper value. This header helps you enforce the cross-site scripting (XSS) protection built into modern browsers.
- Adding the "x-content-type" header with a proper value. This header helps you prevent attacks based on content type shifting.

Example: Notify WebLogic that the load balancer terminated SSL

You can configure your load balancer to perform SSL termination. Often, your backend applications require notification of this action. For example, HTTPS WebLogic e-commerce online transaction processing looks for the `WL-Proxy-SSL` header to confirm that a request came in over SSL. You can use rule sets to add this header at the load balancer listener.

1. Follow the instructions to create a rule set and:
    a. Choose the **Add Request Header** option from the **Action** drop-down list.
    b. Enter `WL-Proxy-SSL` as the **Header** name.
    c. Set the header **Value**:
        - If your load balancer is configured to perform SSL termination, set this value to "true".
        - If the SSL termination point is in the web server where the plug-in operates, set this value to "false".
2. Create a listener, or edit an existing listener, and add the new rule set.

Using the Console

To apply a rule set to a listener, you first create the rule set that contains the rules. The rule set becomes a part of the load balancer's configuration. You can specify the rule set to use when you create or update a listener for the load balancer.

To create a rule set

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking**

and click **Load Balancers**.

2. Choose the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
3. In the **Resources** menu, click **Rule Sets** (if necessary), and then click **Create Rule Set**.
4. In the **Create Path Route Set** dialog box, enter the following:
    - **Name:** Required. Specify a friendly name for the path route set. The name must be unique, and cannot be changed. Avoid entering confidential information.
    - **Rules**
        - **Order:** Optional. If you have multiple rules, you can click the up or down arrows to move the corresponding rule.
        - **Action:** Select the action that the rule applies. Available actions include:
            - **Add Request Header**

              Adds the specified header and value to the incoming request.

              If the specified header is already present, the system replaces it.

              If more than one header with the same name is present, the system removes all of them and adds one header corresponding to the specified header and value.
            - **Extend Request Header**

              Adds the specified prefix or suffix to the incoming request.

              You must provide a prefix value, a suffix value, or both when you choose this action.

              The system does not support this rule for headers with multiple values.
            - **Remove Request Header**

              Removes the specified header.

              If the same header appears more than once in the request, the load balancer removes all occurrences of the specified header.

- **Add Response Header**
  Adds the specified header and value to the outgoing response.
  If the specified header is already present, the system replaces it.
  If more than one header with the same name is present, the system removes all of them and adds one header corresponding to the specified header and value.
- **Extend Response Header**
  Adds the specified prefix or suffix to the incoming request.
  You must provide a prefix value, a suffix value, or both when you choose this action.
  The system does not support this rule for headers with multiple values.
- **Remove Response Header**
  Removes the specified header.
  If the same header appears more than once in the response, the load balancer removes all occurrences of the specified header.



Note

These rules apply only to HTTP or HTTP2 headers.

- **Header:** A header name that conforms to RFC 7230.



Warning

The system does not distinguish between underscore and dash characters in headers. That is, it treats example_header_name and example-header-name as identical. Oracle recommends that you do not rely on underscore or dash characters to uniquely distinguish header names.

- **Value:** (Add rules only.) A header value that conforms to RFC 7230.
- **Prefix:** (Extend rules only.) A character string to add to the beginning of the existing header name. The resulting header must conform to RFC 7230.
- **Suffix:** (Extend rules only.) A character string to add to the end of the existing header name. The resulting header must conform to RFC 7230.

5. (Optional) Click **+ Additional Rule** to create another rule or click the red box to delete an existing rule. You can have up to 20 rules in a set.
6. Click **Create**.

After you create a rule set, the set becomes available for use with the associated load balance. Create or update a listener to apply the rule set.

To update a rule set

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
2. Choose the **Compartment** that contains the load balancer associated with the rule set you want to modify, and then click the load balancer's name.
3. In the **Resources** menu, click **Rule Sets** (if necessary).
4. Click the name of the rule set you want to edit, and then click **Edit Rules**.

5. In the **Edit Rule Set** dialog box, add or remove rules as needed.
   You cannot edit the **Name** field of an existing rule set.
6. Click **Save Changes**.

To remove a rule set from a listener

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
2. Choose the **Compartment** that contains the load balancer associated with the rule set you want to delete, and then click the load balancer's name.
3. In the **Resources** menu, click **Listeners** (if necessary).
4. For the listener you want to edit, click the Actions icon (three dots), and then click **Edit Listener**.
5. In the **Rule Sets** section of the dialog box, click the red box to remove an existing rule set.



Tip

This action removes the rule set from the current listener, but the rule set remains available for application to other listeners on the load balancer.

6. Click **Submit**.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage rule sets:

- CreateRuleSet
- DeleteRuleSet
- GetRuleSet
- ListRuleSets
- UpdateRuleSet

Managing SSL Certificates

This topic is part of the setup and maintenance of a load balancer. For more information about managing load balancers, see Managing a Load Balancer.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: For a typical policy that gives access to load balancers and their components, see Let network admins manage load balancers.

Also, be aware that a policy statement with `inspect load-balancers` gives the specified group the ability to see *all* information about the load balancers. For more information, see Details for Load Balancing.

If you're new to policies, see Getting Started with Policies and Common Policies.

Working with SSL Certificates

To use SSL with your load balancer, you must add one or more certificate bundles to your system. The certificate bundle you upload includes the public certificate, the corresponding private key, and any associated Certificate Authority (CA) certificates. For the easiest workflow, upload the certificate bundles you want to use *before* you create the listeners or backend sets you want to associate them with.

Load balancers commonly use single domain certificates. However, load balancers with listeners that include request routing configuration might require a subject alternative name (SAN) certificate (also called multi-domain certificate) or a wildcard certificate. The Load Balancing service supports each of these certificate types.

Important

- The Load Balancing service does not generate SSL certificates. It can only import an existing certificate that you already own. The certificate can be one issued by a vendor, such as Verisign or GoDaddy. You can also use a self-signed certificate that you generate with an open source tool, such as OpenSSL or Let's Encrypt. Refer to the corresponding tool's documentation for instructions on how to generate a self-signed certificate.
- If you submit a self-signed certificate for backend SSL, you must submit the same certificate in the corresponding CA Certificate field.

Oracle Cloud Infrastructure accepts x.509 type certificates in PEM format only. The following is an example PEM encoded certificate:

```
-----BEGIN CERTIFICATE-----
<Base64_encoded_certificate>
-----END CERTIFICATE-----
```

Converting to PEM format

If you receive your certificates and keys in formats other than PEM, you must convert them before you can upload them to the system. You can use OpenSSL to convert certificates and keys to PEM format. The following example commands provide guidance.

CERTIFICATE OR CERTIFICATE CHAIN FROM DER TO PEM

```
openssl x509 -inform DER -in <certificate_name>.der -outform PEM -out <certificate_name>.pem
```

PRIVATE KEY FROM DER TO PEM

```
openssl rsa -inform DER -in <private_key_name>.der -outform PEM -out <private_key_name>.pem
```

CERTIFICATE BUNDLE FROM PKCS#12 (PFX) TO PEM

```
openssl pkcs12 -in <certificate_bundle_name>.p12 -out <certificate_bundle_name>.pem -nodes
```

CERTIFICATE BUNDLE FROM PKCS#7 TO PEM

```
openssl pkcs7 -in <certificate_bundle_name>.p7b -print_certs -out <certificate_bundle_name>.pem
```

Uploading Certificate Chains

If you have multiple certificates that form a single certification chain, you must include all relevant certificates in one file before you upload them to the system. The following example of a certificate chain file includes four certificates:

```
-----BEGIN CERTIFICATE-----
<Base64_encoded_certificate>
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
<Base64_encoded_certificate>
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
<Base64_encoded_certificate>
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
<Base64_encoded_certificate>
-----END CERTIFICATE-----
```

Submitting Private Keys



Tip

Oracle recommends a minimum length of 2048 bits for your RSA private key.

If your private key submission returns an error, the three most common reasons are:

- You provided an incorrect passphrase.
- Your private key is malformed.
- The system does not recognize the encryption method used for your key.

PRIVATE KEY CONSISTENCY

If you receive an error related to the private key, you can use OpenSSL to check its consistency:

```
openssl rsa -check -in <private_key>.pem
```

This command verifies that the key is intact, the passphrase is correct, and the file contains a valid RSA private key.

DECRYPTING A PRIVATE KEY

If the system does not recognize the encryption technology used for your private key, decrypt the key. Upload the unencrypted version of the key with your certificate bundle. You can use OpenSSL to decrypt a private key:

```
openssl rsa -in <private_key>.pem -out <decrypted_private_key>.pem
```

Updating an Expiring Certificate

To ensure consistent service, you must update (rotate) expiring certificates:

1. Update your client or backend server to work with a new certificate bundle.



Note

The steps to update your client or backend server are unique to your system.

2. Upload the new SSL certificate bundle to the load balancer

a. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
b. Click the name of the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
c. Click the load balancer you want to configure.
d. In the **Resources** menu, click **Certificates**, and then click **Add Certificate**.
e. In the **Add Certificate** dialog box, enter the following:
    - **Certificate Name:** Required. Specify a friendly name for the certificate bundle. It must be unique within the load balancer, and it cannot be changed in the Console. (It can be changed using the API.) Avoid entering confidential information.
    - **Certificate:** Optional. (Required for SSL termination.) Paste the certificate, in PEM format, in this field.
    - **CA Certificate:** Optional. (Recommended for backend SSL termination configurations.) Paste the Certificate Authority certificate (root CA certificate) in this field.
    - **Private Key:** Optional. (Required for SSL termination.) Paste the private key for the certificate in this field.
    - **Passphrase:** Optional. Specify a passphrase.
f. Click **Add Certificate**.

3. Edit listeners or backend sets (as needed) so they use the new certificate bundle

EDITING A LISTENER:

a. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.

b. Choose the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
c. In the **Resources** menu, click **Listeners** (if necessary).
d. For the listener you want to edit, click the Actions icon (three dots), and then click **Edit Listener**.
e. In the **Certificate Name** drop-down list, choose the new certificate bundle.
f. Click **Submit**.

EDITING A BACKEND SET:



Warning

Updating the backend set temporarily interrupts traffic and can drop active connections.

a. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
b. Click the name of the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
c. In the **Resources** menu, click **Backend Sets** (if necessary).
d. Click the name of the backend set you want to edit.
e. Click **Edit Backend Set**.
f. In the **Certificate Name** drop-down list, choose the new certificate bundle.
g. Click **Submit**.

4. (Optional) Remove the expiring SSL certificate bundle



Important

You cannot delete an SSL certificate bundle that is associated with a listener or backend set. Remove the bundle from any additional listeners or backend sets before deleting.

a. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
b. Click the name of the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
c. Click the load balancer you want to configure.
d. In the **Resources** menu, click **Certificates**.
e. For the certificate you want to delete, click the Actions icon (three dots), and then click **Delete**.
f. Confirm when prompted.

Configuring SSL Handling

With Oracle Cloud Infrastructure Load Balancing, you can:

- Terminate SSL at the load balancer. This configuration is *frontend SSL*. Your load balancer can accept encrypted traffic from a client. There is no encryption of traffic between the load balancer and the backend servers.
- Implement SSL between the load balancer and your backend servers. This configuration is *backend SSL*. Your load balancer does not accept encrypted traffic from client servers. Traffic between the load balancer and the backend servers is encrypted.
- Implement end to end SSL. Your load balancer can accept SSL encrypted traffic from clients and encrypts traffic to the backend servers.

Terminating SSL at the Load Balancer

To terminate SSL at the load balancer, you must create a listener at a port such as 443, and then associate an uploaded certificate bundle with the listener.

Implementing Backend SSL

To implement SSL between the load balancer and your backend servers, you must associate an uploaded certificate bundle with the backend set.



Tip

- If you want to have more than one backend server in the backend set, sign your backend servers with an intermediate CA certificate. The intermediate CA certificate must be included as part of the certificate bundle.
- Your backend services must be able to accept and terminate SSL.

Implementing End to End SSL

To implement end to end SSL, you must associate uploaded certificate bundles with both the listener and the backend set.

Using the Console

To upload an SSL certificate bundle to your load balancing system

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.

2. Click the name of the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
3. Click the load balancer you want to configure.
4. In the **Resources** menu, click **Certificates**, and then click **Add Certificate**.
5. In the **Add Certificate** dialog box, enter the following:
    - **Certificate Name:** Required. Specify a friendly name for the certificate bundle. It must be unique within the load balancer, and it cannot be changed in the Console. (It can be changed using the API.) Avoid entering confidential information.
    - **Certificate:** Optional. (Required for SSL termination.) Paste the certificate, in PEM format, in this field.

      Important

      If you submit a self-signed certificate for backend SSL, you must submit the same certificate in the corresponding CA Certificate field.

    - **CA Certificate:** Optional. (Recommended for backend SSL termination configurations.) Paste the Certificate Authority certificate (root CA certificate) in this field.
    - **Private Key:** Optional. (Required for SSL termination.) Paste the private key for the certificate in this field.
    - **Passphrase:** Optional. Specify a passphrase.
6. Click **Add Certificate**.

To delete an SSL certificate bundle from your load balancing system

> **Important**
>
> You cannot delete an SSL certificate bundle that is associated with a listener or backend set. Remove the bundle from any listeners or backend sets before deleting.

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
2. Click the name of the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
3. Click the load balancer you want to configure.
4. In the **Resources** menu, click **Certificates**.
5. For the certificate you want to delete, click the Actions icon (three dots), and then click **Delete**.
6. Confirm when prompted.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage load balancer certificates:

- CreateCertificate
- DeleteCertificate
- ListCertificates

Editing Health Check Policies

This topic describes how to modify health check policies for a backend set.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: For a typical policy that gives access to load balancers and their components, see Let network admins manage load balancers.

Also, be aware that a policy statement with `inspect load-balancers` gives the specified group the ability to see *all* information about the load balancers. For more information, see Details for Load Balancing.

If you're new to policies, see Getting Started with Policies and Common Policies.

Working with Health Check Policies

A health check is a test to confirm the availability of backend servers. A health check can be a request or a connection attempt. Based on a time interval you specify, the load balancer applies the health check policy to continuously monitor backend servers. If a server fails the health check, the load balancer takes the server temporarily out of rotation. If the server subsequently passes the health check, the load balancer returns it to the rotation.

You configure your health check policy when you create a backend set. You can configure TCP-level or HTTP-level health checks for your backend servers.

- TCP-level health checks attempt to make a TCP connection with the backend servers and validate the response based on the connection status.
- HTTP-level health checks send requests to the backend servers at a specific URI and validate the response based on the status code or entity data (body) returned.

The service provides application-specific health check capabilities to help you increase availability and reduce your application maintenance window.



Important

Configure your health check protocol to match your application or service.

If you run an HTTP service, be sure to configure an HTTP-level health check. If you run a TCP-level health check against an HTTP service, you might not get an accurate response. The TCP handshake can succeed and indicate that the service is up even when the HTTP service is incorrectly configured or having other issues. Although the health check appears good, customers might experience transaction failures. For example:

- The backend HTTP service has issues and returns 5XX messages. An HTTP health check catches the message and marks the service as down. In this case, a TCP health check handshake succeeds and marks the service as healthy, even though the HTTP service is not usable.
- The backend HTTP service responds with 4XX messages due to authorization issues or no configured content. A TCP health check does not catch these errors.

Health Status

The Load Balancing service provides health status indicators that use your health check policies to report on the general health of your load balancers and their components. You can see health status indicators on the Console *List* and *Details* pages for load balancers, backend

sets, and backend servers. You also can use the Load Balancing API to retrieve this information.

There are four levels of health status indicators. The general meaning of each level is:

OK (GREEN)

No attention required.

The resource is functioning as expected.

WARNING (YELLOW)

Some reporting entities require attention.

The resource is not functioning at peak efficiency or the resource is incomplete and requires further work.

CRITICAL (RED)

Some or all reporting entities require immediate attention.

The resource is not functioning or unexpected failure is imminent.

UNKNOWN (GRAY)

Health status cannot be determined.

The resource is not responding or is in transition and might resolve to another status over time.

The precise meaning of each level differs among the following components:

- Load balancers
- Backend sets
- Backend servers

Using Health Status

At the highest level, load balancer health reflects the health of its components. The health status indicators provide information you might need to drill down and investigate an existing

issue. Some common issues that the health status indicators can help you detect and correct include:

A HEALTH CHECK IS MISCONFIGURED.

In this case, all the backend servers for one or more of the affected listeners report as unhealthy. If your investigation finds that the backend servers do not have problems, then a backend set probably includes a misconfigured health check.

A LISTENER IS MISCONFIGURED.

All the backend server health status indicators report **OK**, but the load balancer does not pass traffic on a listener.

The listener might be configured to:

- Listen on the wrong port.
- Use the wrong protocol.
- Use the wrong policy.

If your investigation shows that the listener is not at fault, check the security list configuration.

A SECURITY LIST IS MISCONFIGURED.

Health status indicators help you diagnose two cases of misconfigured security lists:

- All entity health status indicators report **OK**, but traffic does not flow (as with misconfigured listeners). If the listener is not at fault, check the security list configuration.
- All entity health statuses report as unhealthy. You have checked your health check configuration and your services run properly on your backend servers.
  In this case, your security lists might not include the IP range for the source of the health check requests. You can find the health check source IP on the *Details* page for each backend server. You can also use the API to find the IP in the `sourceIpAddress` field of the HealthCheckResult object.



Note

Source IP

The source IP for health check requests comes from a Compute instance managed by the Load Balancing service.

ONE OR MORE OF THE BACKEND SERVERS REPORTS AS UNHEALTHY.

A backend server might be unhealthy or the health check might be misconfigured. To see the corresponding error code, check the **status** field on the backend server's *Details* page. You can also use the API to find the error code in the `healthCheckStatus` field of the HealthCheckResult object.

OTHER CASES IN WHICH HEALTH STATUS MIGHT PROVE HELPFUL INCLUDE:

- VCN security lists block traffic.
- Compute instances have misconfigured route tables.

Health Status Limitations

Health status is updated every three minutes. No finer granularity is available.

Health status does not provide historical health data.

Using the Console

You create your health check tests when you create a backend set.

To edit an existing health check policy

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.

2. Click the name of the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
3. In the **Resources** menu, click **Backend Sets** (if necessary).
4. For the backend set you want to modify, click the Actions icon (three dots), and then click **View Backend Set Details**.
5. Click **Update Health Check**.
6. In the **Health Check** section, specify the test parameters to confirm the health of backend servers.



Tip

All parameters are required when updating an existing health check policy.

- **Protocol:** Required. Specify the protocol to use, either HTTP or TCP.



Important

Configure your health check protocol to match your application or service.

- **Port:** Required. Specify the backend server port against which to run the health check.



Tip

You can enter the value '0' to have the health check use the backend server's traffic port.

- **URL Path (URI):** (HTTP only) Optional. Specify a URL endpoint against which to run the health check.
- **Interval in ms:** Optional. Specify how frequently to run the health check, in milliseconds. Default is 10000 (10 seconds).
- **Timeout in ms:** Optional. Specify the maximum time in milliseconds to wait for a reply to a health check. A health check is successful only if a reply returns within this timeout period. Default is 3000 (3 seconds).
- **Number of retries:** Optional. Specify the number of retries to attempt before a backend server is considered "unhealthy". Default is 3.
- **Status Code:** (HTTP only) Required. Specify the status code a healthy backend server must return.
- **Response Body Regex:** (HTTP only) Required. Provide a regular expression for parsing the response body from the backend server. The system treats a blank entry here as the value ".*".



Tip

Health checks require all fields to match. Your status code and response body both must match, as specified.

7. Click **Save**.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use this API operation to edit a backend set's health check policy:

UpdateBackendSet

Use these API operations to retrieve health status information:

- GetBackendHealth
- GetBackendSetHealth
- GetLoadBalancerHealth
- ListLoadBalancerHealths

Viewing the State of a Work Request

This topic describes how to view the state of work requests associated with a given load balancer.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: For a typical policy that gives access to load balancers and their components, see Let network admins manage load balancers.

Also, be aware that a policy statement with `inspect load-balancers` gives the specified group the ability to see *all* information about the load balancers. For more information, see Details for Load Balancing.

If you're new to policies, see Getting Started with Policies and Common Policies.

Monitoring Work Requests

Many of the Oracle Cloud Infrastructure Load Balancing service requests do not take effect immediately. In these cases, the request spawns an asynchronous workflow for fulfillment. To provide visibility for in-progress workflows, the Load Balancing service creates a work request object. Because some operations depend on the completion of other operations, you

must monitor each operation's work request and confirm it has succeeded before proceeding to the next operation. For example, if you want to create a backend set and add a backend server to the new set, you first must create the backend set. After that operation completes, you can add the backend server. If you try to add a backend server before the backend set creation completes, the system cannot ensure that the request to add the server succeeds. You can monitor the request to add a backend set to determine when that workflow is complete, and then add the backend server.

The work request states are:

ACCEPTED

The request is in the work request queue to be processed.

IN PROGRESS

A work request record exists for the specified request, but there is no associated WORK_COMPLETED record.

SUCCEEDED

A work request record exists for this request and an associated WORK_COMPLETED record has the state SUCCEEDED.

FAILED

A work request record exists for this request and an associated WORK_COMPLETED record has the state FAILED.

Using the Console

The Oracle Cloud Infrastructure Console consumes the REST API and is subject to the same considerations as any Oracle Cloud Infrastructure client. You can view the state of a load balancing work request in the Console:

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.

2. Click the name of the **Compartment** that contains the load balancer you want to review, and then click the load balancer's name.
3. In the **Resources** menu, click **Work Requests**. The status of all work requests appears on the page.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these operations to monitor the state of work requests:

- ListWorkRequests
- GetWorkRequest

Load Balancing Metrics

You can monitor the health, capacity, and performance of your load balancers by using metrics, alarms, and notifications.

This topic describes the metrics emitted by the Load Balancing service in the `oci_lbaas` metric namespace.

Resources: Load balancers, listeners, and backend sets.

Overview of the Load Balancing Service Metrics

Your load balancer acts as an intermediary for data traffic between clients and your application servers. Clients send requests to your load balancer and the load balancer distributes the requests to your backend servers according to rules you establish. See the diagram in Overview of Load Balancing for a high-level view of a simple public load balancing system configuration.

The Load Balancing service metrics help you measure the number and type of connections, and quantity of data managed by your load balancer. You can use metrics data to diagnose

and troubleshoot load balancer and client issues. The metrics also help you analyze the HTTP responses returned by the servers in your backend set.

To view a default set of metrics charts in the Console, navigate to the load balancer or backend set you're interested in, and then click **Metrics**. You also can use the Monitoring service to create custom queries.

Prerequisites

- IAM policies: To monitor resources, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. The policy must give you access to the monitoring services as well as the resources being monitored. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in. For more information on user authorizations for monitoring, see the Authentication and Authorization section for the related service: Monitoring or Notifications.
- The metrics listed on this page are automatically available for any load balancer, listener, and backend set you create. You do not need to enable monitoring on the resource to get these metrics.

Available Metrics: oci_lbaas

Load Balancing service metrics include the following dimensions:

AVAILABILITYDOMAIN

The availability domain in which the load balancer resides.

BACKENDSETNAME

The name of the backend set to which the metrics apply.

LBCOMPONENT

The load balancer component to which the metrics apply.

Valid metrics for the Load Balancing service vary among the three **lbComponent** dimension values:

- **Backendset**
- **Listener**
- **Loadbalancer**

The tables on this page describe which data is valid for each of these dimension values. If you choose a metric that does not apply to the specified dimension value, the metric returns no data.

LBHOSTID

A unique ID that represents the current load balancer host. This ID is subject to change.

LISTENERNAME

The name of the listener to which the metrics apply.

REGION

The region in which the load balancer resides.

RESOURCEID

The OCID of the resource to which the metrics apply.

Metrics for the lbComponent Dimension Value "Backendset"

Metric	Metric Display Name	Unit	Description	Dimensions
ActiveConnections	**Active Connections**	count	The number of active connections from the load balancer to all backend servers.	availabilityDomain backendSetName lbComponent lbHostId region resourceId
BackendServers	**Backend Servers**	count	The number of backend servers in the backend set.	
BackendTimeouts	**Backend Timeouts**	count	The number of timeouts across all backend servers.	
BytesReceived	**Bytes Received**	bytes	The number of bytes received across all backend servers.	

Metric	Metric Display Name	Unit	Description	Dimensions
BytesSent	**Bytes Sent**	bytes	The number of bytes sent across all backend servers.	
ClosedConnections	**Closed Connections**	count	The number of connections closed between the load balancer and backend servers.	
HttpRequests	**Inbound Requests**	count	The number of incoming client requests to the backend set.	
HttpResponses	**Responses**	count	The number of HTTP responses across all backend servers.	

Metric	Metric Display Name	Unit	Description	Dimensions
HttpResponses200	**HTTP 200 Responses**	count	The number of HTTP 200 responses received from backend servers.	
HttpResponses2xx	**HTTP 2xx Responses**	count	The number of HTTP 2xx responses received from backend servers.	
HttpResponses3xx	**HTTP 3xx Responses**	count	The number of HTTP 3xx responses received from backend servers.	
HttpResponses4xx	**HTTP 4xx Responses**	count	The number of HTTP 4xx responses received from backend servers.	

Metric	Metric Display Name	Unit	Description	Dimensions
HttpResponses502	**HTTP 502 Responses**	count	The number of HTTP 502 responses received from backend servers.	
HttpResponses504	**HTTP 504 Responses**	count	The number of HTTP 504 responses received from backend servers.	
HttpResponses5xx	**HTTP 5xx Responses**	count	The number of HTTP 5xx responses received from backend servers.	
InvalidHeaderResponses	**Invalid Header Responses**	count	The number of invalid header responses across all backend servers.	

Metric	Metric Display Name	Unit	Description	Dimensions
KeepAliveConnections	**Keep-alive Connections**	count	The number of keep-alive connections.	
ResponseTimeFirstByte	**Average Response Time (TCP only)**	ms	Average time to the first byte of response from backend servers. TCP only.	
ResponseTimeHttpHeader	**Average Response Time (HTTP only)**	ms	Average response time of backend servers. HTTP only.	
UnhealthyBackendServers	**Unhealthy Backend Servers**	count	The number of unhealthy backend servers in the backend set.	

Metrics for the lbComponent Dimension Value"Loadbalancer"

Metric	Metric Display Name	Unit	Description	Dimensions
AcceptedConnections	**Accepted Connections**	count	The number of connections accepted by the load balancer.	availabilityDomain lbComponent lbHostId region resourceId
AcceptedSSLHandshake	**Accepted SSL Handshakes**	count	The number of accepted SSL handshakes.	
ActiveConnections	**Active Connections**	count	The number of active connections from clients to the load balancer.	
ActiveSSLConnections	**Active SSL Connections**	count	The number of active SSL connections.	
BytesReceived	**Bytes Received**	bytes	The number of bytes received by the load balancer.	

Metric	Metric Display Name	Unit	Description	Dimensions
BytesSent	**Bytes Sent**	bytes	The number of bytes sent by the load balancer.	
FailedSSLClientCertVerify	**Failed Client SSL Cert Verifications**	count	The number of failed client SSL certificate verifications.	
FailedSSLHandshake	**Failed SSL Handshakes**	count	The number of failed SSL handshakes.	
HandledConnections	**Handled Connections**	count	The number of connections handled by the load balancer.	
HttpRequests	**Inbound Requests**	count	The number of incoming client requests to the load balancer.	

Metrics for the lbComponent Dimension Value "Listener"

Metric	Metric Display Name	Unit	Description	Dimensions
HttpResponses200	**HTTP 200 Responses**	count	The number of HTTP 200 responses received from backend sets.	availabilityDomain lbComponent lbHostId listenerName region resourceId
HttpResponses2xx	**HTTP 2xx Responses**	count	The number of HTTP 2xx responses received from backend sets.	
HttpResponses3xx	**HTTP 3xx Responses**	count	The number of HTTP 3xx responses received from backend sets.	
HttpResponses4xx	**HTTP 4xx Responses**	count	The number of HTTP 4xx responses received from backend sets.	
HttpResponses502	**HTTP 502 Responses**	count	The number of HTTP 502 responses received from backend sets.	
HttpResponses504	**HTTP 504 Responses**	count	The number of HTTP 504 responses received from backend sets.	
HttpResponses5xx	**HTTP 5xx Responses**	count	The number of HTTP 5xx responses received from backend sets.	
HttpResponses	**Responses**	count	The number of incoming responses received from backend sets.	

Using the Console

To view default metric charts for a single load balancer

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
2. Choose the **Compartment** that contains the load balancer you want to view, and then click the load balancer's name.
3. In the **Resources** menu, click **Metrics** (if necessary).
   The **Metrics** page displays a default set of charts for the current load balancer.

For more information about monitoring metrics and using alarms, see Monitoring Overview. For information about notifications for alarms, see Notifications Overview.

To view default metric charts for multiple load balancers

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Service Metrics**.
2. For **Metric Namespace**, select **oci_lbaas**.
   The **Service Metrics** page displays a default set of charts for the selected metric namespace. For more information about the emitted metrics, see the foregoing table. You can also use the Monitoringservice to create custom queries.

For more information about monitoring metrics and using alarms, see Monitoring Overview. For information about notifications for alarms, see Notifications Overview.

Using the API

Use the following APIs for monitoring:

- Monitoring API for metrics and alarms
- Notifications API for notifications (used with alarms)

CHAPTER 16 Marketplace

This chapter explains how to work with listings in the Oracle Cloud Infrastructure Marketplace.

Overview of Marketplace

Oracle Cloud Infrastructure Marketplace is an online store that offers applications specifically for customers of Oracle Cloud Infrastructure. In the Oracle Cloud Infrastructure Marketplace catalog, you can find images from partners that you can deploy on an Oracle Cloud Infrastructure Compute instance. Images are templates of virtual hard drives that determine the operating system and software to run on an instance. With an image from a partner, you have a more streamlined way of getting started with their software.

Resource Identifiers

Most types of Oracle Cloud Infrastructure resources have a unique, Oracle-assigned identifier called an Oracle Cloud ID (OCID). For information about the OCID format and other ways to identify your resources, see Resource Identifiers.

Ways to Access Oracle Cloud Infrastructure

You can access Oracle Cloud Infrastructure using the Console (a browser-based interface) or the REST API. However, you cannot specifically access Marketplace using the REST API. Marketplace does not have SDK support. Instructions for the Console are included in topics throughout this guide.

To access the Console, you must use a supported browser. Oracle Cloud Infrastructure supports the latest desktop versions of Google Chrome, Microsoft Edge, Internet Explorer 11, Safari, Firefox, and Firefox ESR. Note that private browsing mode is not supported for Firefox, Internet Explorer, or Edge. Mobile browsers are not supported.

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

Working with Listings

This topic describes how to view listings and how to launch an instance based on an image from a listing. By default, Oracle Cloud Infrastructure Marketplace displays all listings in its catalog.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

If you're new to policies, see Getting Started with Policies and Common Policies.



Tip

For administrators, the following policies provide access to Marketplace. These policies specify different target groups as examples, but you can specify the same target group (for example, MarketplaceUsers) when applying the policies in your own tenancy.

- The policy Let users list and subscribe to images from the Partner Image catalog gives the specified group the ability to list, read, and use Marketplace listings.
- The policy Let users launch Compute instances gives the specified group general access to managing instances and images, along with the required level of access to attach existing block volumes to the instances.
- If you need to write more restrictive policies for Marketplace, see Details for the Core Services.

Using the Console

To view a listing's details

1. Open the navigation menu. Under the **Solutions, Platform and Edge** group, go to **Marketplace**.
2. Click the listing that you're interested in.
3. Marketplace displays the listing overview by default. To view other details, do the following:

- To view information about the publisher, click **Provider**.
- To view other listings from the same publisher, click **More Apps**.
- To view instructions for using the instance that you create from the listing, click **Usage Information**.

To launch an instance based on a partner image



Tip

When you create an instance, several other resources are involved (for example, an image, a cloud network, or a subnet). Those other resources can be in the same compartment with the instance or in other compartments. You must have the required level of access to each of the compartments involved in order to launch the instance. This is also true when you attach a volume to an instance; they don't have to be in the same compartment, but if they're not, you need the required level of access to each of the compartments.

1. Open the navigation menu. Under the **Solutions, Platform and Edge** group, go to **Marketplace**.
2. Click the listing that you're interested in.
3. Review the **Usage Information** tab and ensure you understand what you will need to deploy and to access the instance after you launch it. Both Linux and Windows instances require a cloud network to launch the instance into. For more information, see Overview of Networking. Depending on the type of instance, to access it, you might need an SSH key pair or a security list that enables Remote Desktop Protocol. For more information, see Managing Key Pairs on Linux Instances and To enable RDP access .

4. When you are ready, accept the terms of use, and then click **Launch Instance**.
5. Under **Package Version**, click the image you want to install. By default, the menu displays the latest version.
6. Under **Compartment**, click the name of the compartment where you want to launch the instance.
7. Click **Launch Instance**
8. To finish launching the instance, follow the instructions in Creating an Instance.

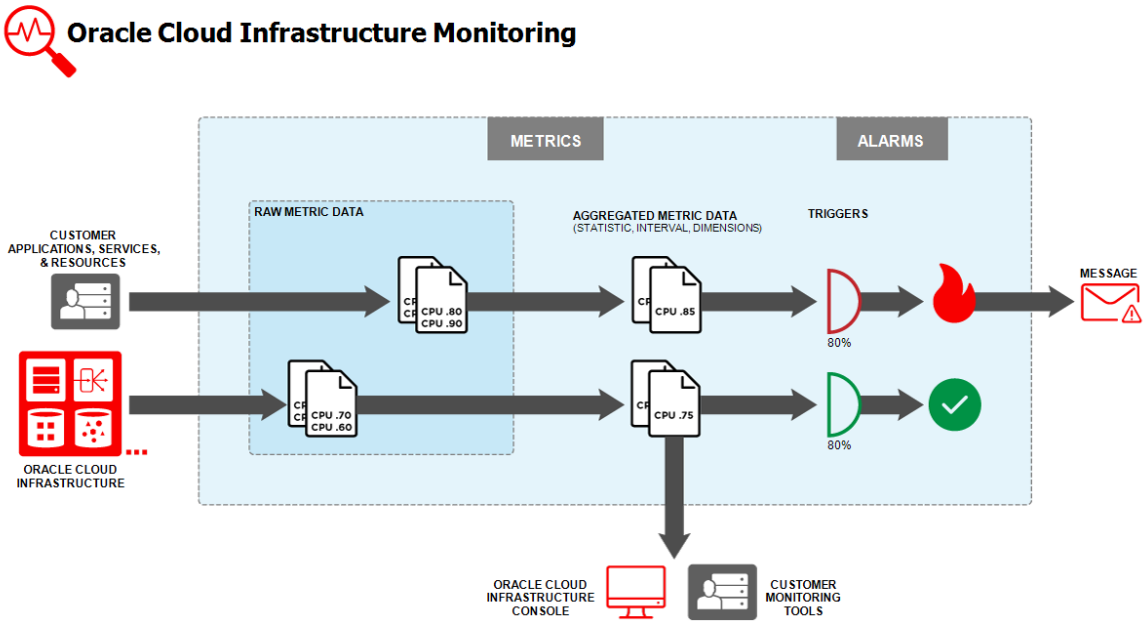
The information you need to connect to an instance after you create it might be in the **Usage Information** or the **Related Documents** sections of the listing.

CHAPTER 17 Monitoring

This chapter explains how to actively and passively monitor performance and usage metrics for your resources.

Monitoring Overview

The Oracle Cloud Infrastructure Monitoring service enables you to actively and passively monitor your cloud resources using the Metrics and Alarms features.



Metrics Feature Overview

The Metrics feature relays metric data about the health, capacity, and performance of your cloud resources. A metric is a measurement related to health, capacity, or performance of a given resource. Resources, services, and applications emit metrics to the Monitoring service. Common metrics reflect data related to availability and latency, application uptime and downtime, completed transactions, failed and successful operations, and key performance indicators (KPIs), such as sales and engagement quantifiers.

By querying Monitoring for this data, you can understand how well the systems and processes are working to achieve the service levels you commit to your customers. For example, you can monitor the CPU utilization and disk reads of your Compute instances. You can then use this data to determine when to launch more instances to handle increased load, troubleshoot issues with your instance, or better understand system behavior.

Example Metric: Failure Rate

For application health, one of the common KPIs is failure rate, for which a common definition is the number of failed transactions divided by total transactions. This KPI is usually delivered through application monitoring and management software.

As a developer, you can capture this KPI from your applications using custom metrics. Simply record observations every time an application transaction takes place and then post that data to the Monitoring service. In this case, set up metrics to capture failed transactions, successful transactions, and transaction latency (time spent per completed transaction).

Alarms Feature Overview

The Alarms feature of the Monitoring service works with the Notifications service to notify you when metrics meet alarm-specified triggers. When configured, repeat notifications remind you of a continued firing state at the configured repeat interval. You are also notified when an alarm transitions back to the OK state, or when an alarm is reset.

You can search for alarms using Search-supported attributes. For more information about Search, see Overview of Search.

Search-Supported Attributes for Alarms

For attribute descriptions, see Alarm Reference.

- id
- displayName
- compartmentId
- metricCompartmentId
- namespace
- query
- severity
- destinations
- suppression
- isEnabled
- lifecycleState
- timeCreated
- timeUpdated
- tags

Monitoring Concepts

The following concepts are essential to working with Monitoring.

AGGREGATED DATA

The result of applying a *statistic* and *interval* to a selection of raw *data points* for a given *metric*. For example, you can apply the *statistic* `max` and *interval* `1h` (one hour) to the last 24 hours of raw *data points* for the *metric* `CpuUtilization`. Aggregated data is displayed in default metric charts in the Console. You can also build metric queries for specific sets

of aggregated data. For instructions, see Viewing Default Metric Charts and Building Metric Queries.

ALARM

The *alarm query* to evaluate and the *notification destination* to use when the alarm is in the firing state, along with other alarm properties. For instructions on managing alarms, see Managing Alarms.

ALARM QUERY

The Monitoring Query Language (MQL) expression to evaluate for the *alarm*. An alarm query must specify a *metric*, *statistic*, *interval*, and a *trigger rule* (threshold or absence). The Alarms feature of the Monitoring service interprets results for each returned time series as a Boolean value, where zero represents false and a non-zero value represents true. A true value means that the *trigger rule* condition has been met. For more information, see Building Metric Queries and the query attribute description in the Alarm API reference.

DATA POINT

A timestamp-value pair for the specified *metric*. Example: 2018-05-10T22:19:00Z, 10.4

A data point is either raw or aggregated. Raw data points are posted by the *metric namespace* to the Monitoring service using the PostMetricData operation. The *frequency* of the data points posted varies by *metric namespace*. For example, your custom namespace might send data points for a given *metric* at a 20-second *frequency*.

Aggregated data points are the result of applying a *statistic* and *interval* to raw data points. The *interval* of the aggregated data points is determined by the SummarizeMetricsData request. For example, a request specifying the *statistic* `sum` and *interval* `1h` (one hour) returns a `sum` value for each hour of available raw data points for the given *metric*.

DIMENSION

A qualifier provided in a *metric definition*. Example: Resource identifier (`resourceId`), provided in the definitions of oci_computeagent metrics. Use dimensions to filter or group

metric data. Example dimension name-value pair for filtering by availability domain: `availabilityDomain = "VeBZ:PHX-AD-1"`

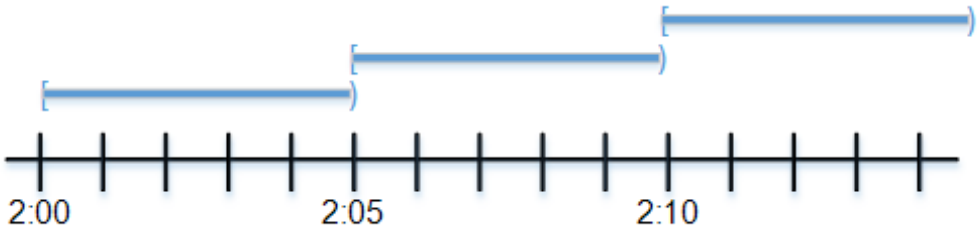
FREQUENCY

The time period between each posted raw *data point* for a given *metric*. (Raw data points are posted by the *metric namespace* to the Monitoring service.) While frequency varies by metric, default service metrics typically have a frequency of 60 seconds (that is, one data point posted per minute). See also *resolution*.

INTERVAL

The time window used to convert the given set of raw *data points*.

The timestamp of the aggregated data point corresponds to the beginning of the time window during which raw data points are assessed. For example, for a five-minute interval, the timestamp "2:05" corresponds to the five-minute time window from 2:05 to 2:09:*n*. During the time window, the value of this aggregated data point dynamically updates. The final value for the aggregated data point is obtained when the last raw data point is assessed. There may be a short delay before the final value is posted.



The following example query specifies a 5-minute interval. `CpuUtilization[5m].max()` For supported values, see Monitoring Query Language (MQL) Reference.

See also *resolution*.

MESSAGE

The content that the Alarms feature of the Monitoring service publishes to topics in the *alarm's* configured *notification destinations*. A message is sent when the *alarm* transitions to another state, such as from "OK" to "FIRING." For more information about messages, see How Monitoring Works.

METADATA

A reference provided in a *metric definition*. Example: unit (bytes), provided in the definition of the oci_computeagent *metric* `DiskBytesRead`. Use metadata to determine additional information about a given metric. For metric definitions, see Supported Services.

METRIC

A measurement related to health, capacity, or performance of a given resource. Example: The oci_computeagent *metric* `CpuUtilization`, which measures usage of a Compute instance. For metric definitions, see Supported Services.



Note

Metric resources do not have OCIDs.

METRIC DEFINITION

A set of references, qualifiers, and other information provided by a *metric namespace* for a given *metric*. For example, the oci_computeagent *metric* `DiskBytesRead` is defined by *dimensions* (such as resource identifier) and *metadata* (specifying bytes for unit) as well as identification of its *metric namespace* (oci_computeagent). Each posted set of *data points* carries this information. Use the ListMetricData API operation to get metric definitions. For metric definitions, see Supported Services.

METRIC NAMESPACE

Indicator of the resource, service, or application that emits the *metric*. Provided in the *metric definition*. For example, the `CpuUtilization` *metric definition* emitted by the

OracleCloudAgent software on Compute instances lists the *metric namespace* `oci_computeagent` as the source of the `CpuUtilization` *metric*. For metric definitions, see Supported Services.

METRIC STREAM

An individual set of *aggregated data* for a *metric*. A stream can be either specific to a single resource or aggregated across all resources in the compartment. Within a metric chart in the Console, each metric stream is represented as a line. By default, metric streams are resource-specific, so the chart displays a line for each resource. If you choose to aggregate all metric streams, then the chart displays one line for all resources.

NOTIFICATION DESTINATION

Protocol and other details for sending *messages* when the *alarm* transitions to another state, such as from "OK" to "FIRING." The details and setup may vary by destination service. For the Notifications service, each destination includes a topic and subscription protocol (such as PagerDuty). For more information about messages, topics, and subscriptions, see Notifications Overview.

ORACLECLOUDAGENT SOFTWARE

Software that allows a Compute instance to post raw *data points* to the Monitoring service. Automatically installed with the latest versions of supported images. See Enabling Monitoring for Compute Instances.

QUERY

The Monitoring Query Language (MQL) expression to evaluate for returning *aggregated data*. The query must specify a *metric*, *statistic*, and *interval*. For more information, see Building Metric Queries.

RESOLUTION

The period between time windows, or the regularity at which time windows shift. For example, use a resolution of `1m` to retrieve aggregations every minute.



Note

For metric queries, the interval you select drives the default resolution of the request, which determines the maximum time range of data returned.

Maximum time range returned for a query

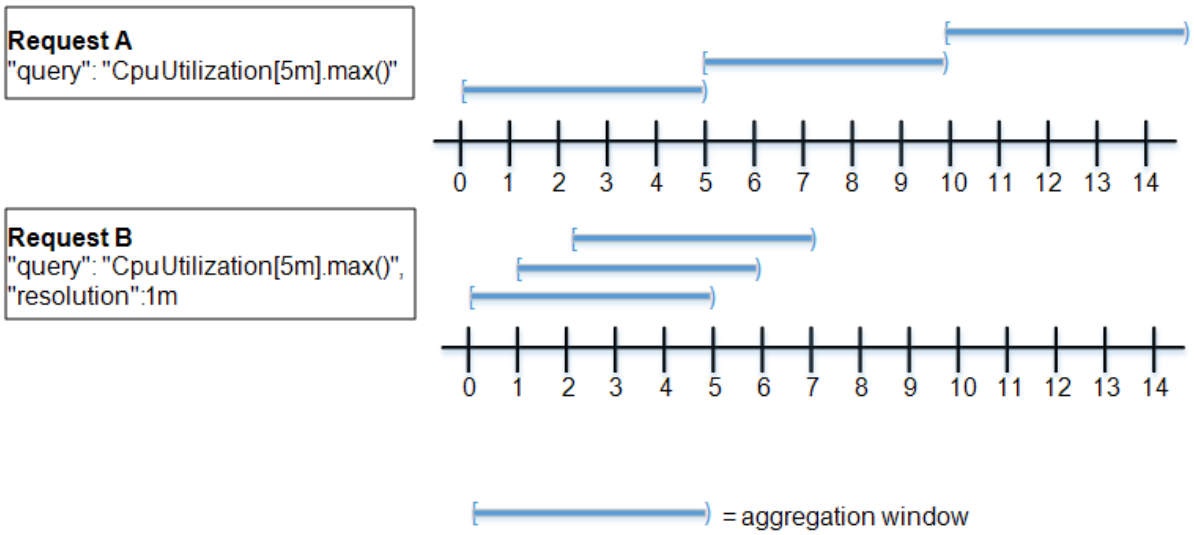
The maximum time range returned for a metric query depends on the resolution. By default, for metric queries, the resolution is the same as the query interval. Following are the maximum time ranges returned for each interval selection available in the Console.

Interval	Default resolution (metric queries)	Maximum time range returned
1h	1 hour	90 days
5m	5 minutes	30 days
1m	1 minute	7 days

For more information about the resolution parameter as used in metric queries, see SummarizeMetricsData.

For alarm queries, the specified interval has no effect on the resolution of the request. The only valid value of the resolution for an alarm query request is `1m`. For more information about the resolution parameter as used in alarm queries, see Alarm.

As shown in the following illustration, *resolution* controls the start time of each aggregation window relative to the previous window while *interval* controls the length of the windows. Both requests apply the statistic `max` to the data within each five-minute window (from the interval), resulting in a single aggregated *data point* representing the highest `CPUutilization` counter for that window. Only the resolution value differs. This resolution changes the regularity at which the aggregation windows shift, or the start times of successive aggregation windows. Request A does not specify a resolution and thus uses the default value equal to the interval (5 minutes). This request's five-minute aggregation windows are thus taken from the sets of data points emitted between minute 0 to minute 4:*n*, minute 5 to minute 9:*n*, and so forth. Request B specifies a 1-minute resolution, so its five-minute aggregation windows are taken from the set of data points emitted every minute from 0 to 4:*n*, 1 to 6:*n*, and so forth.



STATISTIC

The aggregation function applied to the given set of raw *data points*. For supported statistics, see Monitoring Query Language (MQL) Reference.

SUPPRESSION

A configuration to avoid publishing *messages* during the specified time range. Useful for suspending alarm notifications during system maintenance. Each suppression applies to a single alarm. In the Console, you can apply one definition of a suppression to multiple alarms. The result is an individual suppression for each alarm. For instructions on suppressing alarms, see To suppress alarms.

TRIGGER RULE

The condition that must be met for the alarm to be in the firing state. A trigger rule can be based on a threshold or absence of a metric.

How Monitoring Works

The Monitoring service uses metrics to monitor resources and alarms to notify you when these metrics meet alarm-specified triggers.

Metrics are emitted to the Monitoring service by resources as raw data points, or timestamp-value pairs, along with dimensions and metadata. For example, the Compute service (metric namespace "oci_computeagent") posts this data for monitoring-enabled Compute instances. The posted data includes all oci_computeagent metrics, such as CpuUtilization. Metric data posted to the Monitoring service is only presented to you or consumed by the Oracle Cloud Infrastructure features that you enable to use metric data.

When you query a metric, the Monitoring service returns aggregated data according to the specified parameters. You can specify a range (such as the last 24 hours), statistic, and interval. The Console displays one monitoring chart per metric for selected resources. The aggregated data in each chart reflects your selected statistic and interval. API requests can optionally filter by dimension and specify a resolution. API responses include the metric name along with its source compartment and metric namespace. You can feed the aggregated data into a visualization or graphing library.

Metric and alarm data is accessible via the Console, CLI, and API. For retention periods, see Storage Limits.

The Alarms feature of the Monitoring service publishes alarm messages to configured destinations managed by the Notifications service. Each destination is a topic with a set of subscribers. For more information about the Notifications service, see Notifications Overview.

Message types

The reason for sending a message is indicated by its type.

- `OK_TO_FIRING`: The alarm changed from `OK` status to `FIRING` status.
- `FIRING_TO_OK`: The alarm changed from `FIRING` status to `OK` status.
- `REPEAT`: The alarm is maintaining a `FIRING` status and repeat notifications are configured.
- `RESET`: The alarm is not detecting the metric firing; the metric is no longer being emitted. An example scenario for a reset is termination of the resource that was emitting the metric.

Message format and examples

Alarm message format:

Parameter	Description
dedupekey Required	**string** Unique identifier that can be used for de-duplication.
title Required	**string** The alarm's configured display name.
body	**string** The alarm's configured message body.

Parameter	Description
type Required	**string** The reason for sending the notification message. Valid values: See Message types.
severity Required	**string** The highest severity level of the listed alarms. Valid values: `CRITICAL`, `ERROR`, `WARNING`, and `INFO`
timestampEpochMillis Required	**long** The time when the alarm was triggered, in milliseconds since epoch time.
alarmMetadata Required	**array of objects** List of alarms related to this notification message.
version Required	**int** The version of the alarm message format.

alarmMetadata format:

Parameter	Description
id Required	**string** The alarm OCID.
status Required	**string** The alarm state. Valid values: `OK`, `FIRING`

Parameter	Description
severity Required	**string** The alarm severity level. Valid values: `CRITICAL`, `ERROR`, `WARNING`, `INFO`
query Required	**string** The alarm's configured query. `CpuUtilization[1m]{availabilityDomain="cumS:PHX-AD-1"}.absent()`
totalMetricsFiring Required	**int** The number of metric streams represented in this notification message.
dimensions	**array of objects** List of dimension key-value pairs that identify each metric stream. The list is limited to a hundred entries. Empty for an alarm with a status of `OK`.

Example message "High CPU Utilization" for an alarm that is continuing to be in the FIRING state. In this example, the message includes two metric streams: one for "myinstance1" and another for "myinstance2."

```
{
  "dedupeKey": "exampleuniqueID",
  "title": "High CPU Utilization",
  "body": "Follow runbook at http://example.com/runbooks",
  "type": "REPEAT",
  "severity": "CRITICAL",
  "timestampEpochMillis": 1542406320000,
  "alarmMetaData": [
    {
      "id": "ocid1.alarm.oc1.iad.exampleuniqueID",
      "status": "FIRING",
      "severity": "CRITICAL",
```

```
      "query": "CpuUtilization[1m].mean() > 0",
      "totalMetricsFiring": 2,
      "dimensions": [
        {
          "instancePoolId": "Default",
          "resourceDisplayName": "myinstance1",
          "faultDomain": "FAULT-DOMAIN-1",
          "resourceId": "ocid1.instance.oc1.iad.exampleuniqueID",
          "imageId": "ocid1.image.oc1.iad.exampleuniqueID",
          "availabilityDomain": "szYB:US-ASHBURN-AD-1",
          "shape": "VM.Standard2.1",
          "region": "us-ashburn-1"
        },
        {
          "instancePoolId": "Default",
          "resourceDisplayName": "myinstance2",
          "faultDomain": "FAULT-DOMAIN-3",
          "resourceId": "ocid1.instance.oc1.iad.exampleuniqueID",
          "imageId": "ocid1.image.oc1.iad.exampleuniqueID",
          "availabilityDomain": "szYB:US-ASHBURN-AD-1",
          "shape": "VM.Standard2.1",
          "region": "us-ashburn-1"
        }
      ]
    }
  ],
  "version": 1.0
}
```


Availability

Monitoring is currently available in the following regions:

Region Name	Region Location	Region Key
ap-tokyo-1	Asia-Pacific: Tokyo, Japan	NRT
ca-toronto-1	Canada: Toronto	YYZ

Region Name	Region Location	Region Key
eu-frankfurt-1	Europe: Frankfurt, Germany	FRA
uk-london-1	United Kingdom: London	LHR
us-ashburn-1	United States: Ashburn, VA	IAD
us-phoenix-1	United States: Phoenix, AZ	PHX

Supported Services

The following services have resources or components that can emit metrics to Monitoring:

- Block Storage - see Block Volume Metrics
- Compute - see Compute Metrics
- Load Balancing - see Load Balancing Metrics
- Networking - see Networking Metrics
- Notifications - see Notifications Metrics
- Object Storage - see Object Storage Metrics
- Streaming - see Streaming Metrics

Resource Identifiers

Most types of Oracle Cloud Infrastructure resources have a unique, Oracle-assigned identifier called an Oracle Cloud ID (OCID). For information about the OCID format and other ways to identify your resources, see Resource Identifiers.



Note

Metric resources do not have OCIDs.

Ways to Access Monitoring

You can access the Monitoring service using the Console (a browser-based interface) or the REST API. Instructions for the Console and API are included in topics throughout this guide. For a list of available SDKs, see Software Development Kits and Command Line Interface.

Console: To access Monitoring using the Console, you must use a supported browser. Oracle Cloud Infrastructure supports the latest desktop versions of Google Chrome, Microsoft Edge, Internet Explorer 11, Safari, Firefox, and Firefox ESR. Note that private browsing mode is not supported for Firefox, Internet Explorer, or Edge. Mobile browsers are not supported. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring**.

API: To access Monitoring through APIs, use Monitoring API for metrics and alarms and Notifications API for notifications (used with alarms).

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

Administrators: For common policies that give groups access to metrics, see Let users view metric definitions in a compartment and Restrict user access to a specific metric namespace. For a common alarms policy, see Let users view alarms. To authorize resources, such as instances, to make API calls, add the resources to a dynamic group. Use the dynamic group's matching rules to add the resources, and then create a policy that allows that dynamic group

access to metrics. See Let instances make API calls to access monitoring metrics in the tenancy.

Limits on Monitoring

See Monitoring Limits for a list of applicable limits and instructions for requesting a limit increase.

Other limits include the following.

Storage Limits

Item	Time range stored
Metric definitions	14 days
Alarm history entries	90 days

Returned Data Limits (Metrics)

When you query metrics and view metric charts, the returned data is subject to certain limits. Limits information for returned data includes the 100,000 data point maximum and time range maximums (determined by resolution, which relates to interval). See MetricData Reference.

Troubleshooting Limits

If you see an error that the query has exceeded the maximum number of metric streams, then update the query to evaluate a number of metric streams that is within the limit. For example, you can reduce the metric streams by specifying dimensions. You can continue to evaluate all metric streams that were in the original query by spreading the metric streams across multiple queries (or alarms).

Viewing Default Metric Charts

This topic describes how to view metric charts for selected resources or a single resource and create alarms based on queries used for charts. Charts are available using the Console.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Prerequisites

- IAM policies: Viewing metric charts is part of monitoring. To monitor resources, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. The policy must give you access to the monitoring services as well as the resources being monitored. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in. For more information on user authorizations for monitoring, see the Authentication and Authorization section for the related service: Monitoring or Notifications.
- Metrics exist in Monitoring: The resources that you want to monitor must emit metrics to the Monitoring service.
- Compute instances: To emit metrics, Compute instances must be monitoring-enabled. OracleCloudAgent software installation may also be required. For more information, see Enabling Monitoring for Compute Instances.

Working with Default Metric Charts

For background information on metrics in Oracle Cloud Infrastructure, see Metrics Feature Overview. For default metrics by service, see Supported Services.

Default metric charts use predefined service queries. You can select resources of interest and update the interval, statistic, and time range.



Note

Very small or large values are indicated by International System of Units (SI units), such as M for mega (10 to the sixth power). Units correspond to the selected metric and do not change by statistic.

Using the Console

To view default metric charts for all resources

Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Service Metrics**.

The **Service Metrics** page displays the default charts for all resources in the first accessible **Compartment** and **Metric Namespace**. Very small or large values are indicated by International System of Units (SI units), such as M for mega (10 to the sixth power).

Don't see all expected resources or metrics?

- Try a different time range.
- Confirm that the missing resources are emitting metrics. See Enabling Monitoring for Compute Instances.

- Review limits information. Limits information for returned data includes the 100,000 data point maximum and time range maximums (determined by resolution, which relates to interval). See MetricData Reference.

To investigate missing resources or metrics

- Try a different time range.
- Confirm that the missing resources are emitting metrics. See Enabling Monitoring for Compute Instances.
- Review limits information. Limits information for returned data includes the 100,000 data point maximum and time range maximums (determined by resolution, which relates to interval). See MetricData Reference.

To filter results

Filter results to limit the data plotted on the metric chart. For example, filter results to a resource or region of interest.

Filtering is done through selected dimensions; available dimensions vary by metric.

1. View the default metric charts: Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Service Metrics**.
2. To the right of **Dimensions**, click **Add**.
3. In the **Edit dimensions** dialog box, select a **Dimension Name** and **Dimension Value**.

Dimension fields

- **Dimension Name**: A qualifier specified in the metric definition. For example, the dimension `resourceId` is specified in the metric definition for `CpuUtilization`.



Note

Long lists of dimensions are trimmed.

- To view dimensions by name, type one or more characters in the box. A refreshed (trimmed) list shows matching dimension names.
- To retrieve all dimensions for a given metric, use the following API operation: ListMetrics

- **Dimension Value**: The value you want to use for the specified dimension. For example, the resource identifier for your instance of interest.
- **+ Additional dimension**: Adds another name-value pair for a dimension.

4. Click **Done**.
   The default charts show the filtered results of your query.

To select different resources

1. View the default metric charts: Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Service Metrics**.
2. To select resources on a different compartment, select the **Compartment**.
   The default charts update to show results for the selected compartment.
3. To select a specific resource within the selected compartment, filter results by a resource-specific dimension, such as **resourceDisplayName**:
   a. To the right of **Dimensions**, click **Add**.
   b. For **Dimension Name**, select **resourceDisplayName** or other resource-specific dimension.



Note

Long lists of dimensions are trimmed.

- To view dimensions by name, type one or more characters in the box. A refreshed (trimmed) list shows matching dimension names.
- To retrieve all dimensions for a given metric, use the following API operation: ListMetrics

c. For **Dimension Value**, select the value corresponding to the resource you want.

d. Click **Done**.
The default charts update to show filtered results.

To aggregate data from all metric streams

Aggregate all metric streams to view the average. For example, aggregate all metric streams for CPU Utilization to view the average data across all resources. By default, each metric stream is represented by a line in each chart. When aggregated, all metric streams are represented by a single line in each chart.

1. View the default metric charts: Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Service Metrics**.
2. Select **Aggregate Metric Streams**.

To change the time range

1. View the default metric charts: Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Service Metrics**.

2. To select a period of time, such as **Last hour**, click **Start Time** or **End Time**.
3. To enter a time value, click in **Start Time** or **End Time** and then type a value.

To change a chart interval or statistic

1. View the default metric charts: Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Service Metrics**.
2. At the top of the chart you want, select an **Interval** or **Statistic**.
   For supported values, see Monitoring Query Language (MQL) Reference.

To go back to the default charts

1. View the default metric charts: Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Service Metrics**.
2. On the upper right, click **Reset charts**.

To view chart details

Chart details include the query as a Monitoring Query Language (MQL) expression and the names and OCIDs of represented resources.

1. View the default metric charts: Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Service Metrics**.
2. Click the chart you want.
3. To view a list of resources represented in the chart, click the arrow to the left of the query displayed under the chart.
   You can copy the OCID for a resource by clicking **Copy** to the right of the resource OCID.

To share a chart



Note

The person you share the chart with must have the required IAM policies for access to metrics.

On the upper right of the chart you want, go to **Options**, and then click **Copy Chart URL**.

To view a query in Metrics Explorer

On the upper right of the chart you want, go to **Options**, and then click **View Query in Metrics Explorer**.

To copy a query (MQL expression)

On the upper right of the chart you want, go to **Options**, and then click **Copy Query (MQL)**.

To view default metric charts for a single resource

On the page for the resource of interest, under **Resources**, click **Metrics**.

For example, to view metric data for a Compute instance:

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
2. Click the instance you're interested in.
3. On the instance detail page, under **Resources**, click **Metrics**.
   A chart is shown for each metric. For a list of metrics related to Compute instances, see Compute Metrics.

The Console displays the last hour of metric data for the selected resource. A chart is shown for each metric emitted by the selected resource.

For a list of metrics emitted by your resource, see Supported Services.

To create an alarm from a chart query

Follow the instructions for the page on which the query appears: Service Metrics or Metrics Explorer.

Service Metrics page

To create an alarm from a chart query (Service Metrics)

1. View the **Service Metrics** page: Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Service Metrics**.
2. At the top of the chart you're interested in, go to Options, and then select **Create an Alarm on this Query**.
3. On the **Create Alarm** page, under **Define alarm**, add the trigger and fill in or update other alarm settings as needed:

Alarm settings

Basic Mode (default)

By default, this page uses **Basic Mode**, which separates the metric from its dimensions and its trigger rule.

- **Alarm Name**: User-friendly name for the new alarm.
- **Alarm Severity**: The perceived type of response required when the alarm is in the firing state.

- **Alarm Body**: The human-readable content of the notification delivered. Oracle recommends providing guidance to operators for resolving the alarm condition. Consider adding links to standard runbook practices. Example: "High CPU usage alert. Follow runbook instructions for resolution."
- **Tags (optional):** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
- **Metric description**: The metric to evaluate for the alarm condition.
    - **Compartment**: The compartment containing the resources that emit the metrics evaluated by the alarm. The selected compartment is also the storage location of the alarm. By default, the first accessible compartment is selected.
    - **Metric Namespace**: The service or application emitting metrics for the resources that you want to monitor.
    - **Metric Name**: The name of the metric. Only one metric can be specified. Example: **CpuUtilization**

- **Interval**: The aggregation window, or the frequency at which data points are aggregated.

  ### Interval values

  - **1m** - 1 minute
  - **5m** - 5 minutes
  - **1h** - 1 hour

  Note

  For alarm queries, the specified interval has no effect on the resolution of the request. The only valid value of the resolution for an alarm query request is `1m`. For more information about the resolution parameter as used in alarm queries, see Alarm.

- **Statistic**: The aggregation function.

  ### Statistic values

  - **COUNT**- The number of observations received in the specified time period.
  - **MAX** - The highest value observed during the specified time period.

- **MEAN** - The value of Sum divided by Count during the specified time period.
- **MIN** - The lowest value observed during the specified time period.
- **P50** - The value of the 50th percentile.
- **P90** - The value of the 90th percentile.
- **P95** - The value of the 95th percentile.
- **P99** - The value of the 99th percentile.
- **P99.5** - The value of the 99.5th percentile.
- **RATE** - The per-interval average rate of change.
- **SUM** - All values added together.

- **Metric dimensions**: Optional filters to narrow the metric data evaluated.

  Dimension fields

  - **Dimension Name**: A qualifier specified in the metric definition. For example, the dimension `resourceId` is specified in the metric definition for `CpuUtilization`.

    Note

    Long lists of dimensions are trimmed.

    - To view dimensions by name, type one or more characters in the box. A refreshed (trimmed) list shows matching dimension names.
    - To retrieve all dimensions for a given metric, use the following API operation: ListMetrics

  - **Dimension Value**: The value you want to use for the specified dimension. For example, the resource identifier for your instance of interest.
  - **+ Additional dimension**: Adds another name-value pair for a dimension.

- **Trigger rule**: The condition that must be satisfied for the alarm to be in the firing state. The condition can specify a threshold, such as 90% for CPU Utilization, or an absence.

- **Operator**: The operator used in the condition threshold.

  Operator values

  - **greater than**
  - **greater than or equal to**
  - **equal to**
  - **less than**
  - **less than or equal to**
  - **between** (inclusive of specified values)
  - **outside** (inclusive of specified values)
  - **absent**

- **Value**: The value to use for the condition threshold.
- **Trigger Delay Minutes**: The number of minutes that the condition must be maintained before the alarm is in firing state.

Advanced Mode

Click **Advanced Mode** or **Switch to Advanced Mode** to view the alarm query as a Monitoring Query Language (MQL) expression. Edit your query using MQL syntax to aggregate results by group or for additional parameter values. See Monitoring Query Language (MQL) Reference.

- **Alarm Name**: User-friendly name for the new alarm.
- **Alarm Severity**: The perceived type of response required when the alarm is in the firing state.

- **Alarm Body**: The human-readable content of the notification delivered. Oracle recommends providing guidance to operators for resolving the alarm condition. Consider adding links to standard runbook practices. Example: "High CPU usage alert. Follow runbook instructions for resolution."
- **Tags (optional):** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
- **Metric description, dimensions, and trigger rule**: The metric to evaluate for the alarm condition, including dimensions and the trigger rule.
    - **Compartment**: The compartment containing the resources that emit the metrics evaluated by the alarm. The selected compartment is also the storage location of the alarm. By default, the first accessible compartment is selected.
    - **Metric Namespace**: The service or application emitting metrics for the resources that you want to monitor.
    - **Query Code Editor** box: The alarm query as a Monitoring Query Language (MQL) expression.

      Example alarm query:

      ```
      CpuUtilization[1m]{availabilityDomain=AD1}.groupBy(poolId).percentile(0.9) > 85
      ```

      For query syntax and examples, see Working with Metric Queries.
    - **Trigger Delay Minutes**: The number of minutes that the condition must be maintained before the alarm is in firing state.

The chart below the **Define alarm** section dynamically displays the last six hours of emitted metrics according to currently selected fields for the query. Very small or large values are indicated by International System of Units (SI units), such as M for mega (10 to the sixth power).

4. Under **Notifications**, select or create at least one notification destination:

Notifications settings

- **Destinations**:
  - **Destination Service**: The provider of the destination to use for notifications.
    Available options:
    - **Notifications Service**.
  - **Compartment**: The compartment storing the topic to be used for notifications. Can be a different compartment from the alarm and metric. By default, the first accessible compartment is selected.
  - **Topic**: The topic to use for notifications. Each topic supports a subscription protocol, such as PagerDuty.
  - **Create a topic**: Sets up a topic and subscription protocol in the selected compartment, using the specified destination service.
    - **Topic Name**: User-friendly name for the new topic. Example: "Operations Team " for a topic used to notify operations staff of firing alarms.
    - **Topic Description**: Description of the new topic.
    - **Subscription Protocol**: Medium of communication to use for the new topic. Configure your subscription for the protocol you want:

Email subscription

- **Subscription protocol**: Select **Email**.
- **Email Addresses**: Type an email address.

PagerDuty subscription

- **Subscription Protocol**: Select **HTTPS (PagerDuty)**.
- **PagerDuty Integration URL**: Type (or copy and paste) the *integration key* portion of the URL for your PagerDuty subscription. (The other portions of the URL are hard-coded.)
  For information on setting up and retrieving your integration key, see the PagerDuty documentation.
  Example integration key: Your68a4Integration4dd8Key12f4f6
  Example integration URL: https://events.pagerduty.com/integration/Your68a4Integration4dd8Key12f4f6/enqueue

◦ **+ Additional destination service**: Adds another destination service and topic to use for notifications.



Note

Each alarm is limited to one destination per supported destination service.

- **Repeat Notification?**: While the alarm is in the firing state, resends notifications at the specified interval.

- **Notification Interval**: The period of time to wait before resending the notification.
- **Suppress Notifications**: Sets up a suppression time window during which to suspend evaluations and notifications. Useful for avoiding alarm notifications during system maintenance periods.
  - **Suppression Description**
  - **Start Time**
  - **End Time**

5. If you want to disable the new alarm, clear **Enable This Alarm?**.
6. Click **Save alarm**.
   The new alarm is listed on the **Alarm Definitions** page.
   For more information about alarms, see Alarms Feature Overview.

Metrics Explorer page

To create an alarm from a chart query (Metrics Explorer)

1. View the **Metrics Explorer** page: Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Metrics Explorer**.
2. If necessary, open the query for editing: Click the **Edit query** icon.
3. Click **Create Alarm**.
4. On the **Create Alarm** page, under **Define alarm**, add the trigger and fill in or update other alarm settings as needed:

Alarm settings

Basic Mode (default)

By default, this page uses **Basic Mode**, which separates the metric from its dimensions and its trigger rule.

- **Alarm Name**: User-friendly name for the new alarm.
- **Alarm Severity**: The perceived type of response required when the alarm is in the firing state.
- **Alarm Body**: The human-readable content of the notification delivered. Oracle recommends providing guidance to operators for resolving the alarm condition. Consider adding links to standard runbook practices. Example: "High CPU usage alert. Follow runbook instructions for resolution."
- **Tags (optional):** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
- **Metric description**: The metric to evaluate for the alarm condition.
    - **Compartment**: The compartment containing the resources that emit the metrics evaluated by the alarm. The selected compartment is also the storage location of the alarm. By default, the first accessible compartment is selected.
    - **Metric Namespace**: The service or application emitting metrics for the resources that you want to monitor.
    - **Metric Name**: The name of the metric. Only one metric can be specified. Example: **CpuUtilization**

- **Interval**: The aggregation window, or the frequency at which data points are aggregated.

  ### Interval values

  - **1m** - 1 minute
  - **5m** - 5 minutes
  - **1h** - 1 hour

  Note

  For alarm queries, the specified interval has no effect on the resolution of the request. The only valid value of the resolution for an alarm query request is `1m`. For more information about the resolution parameter as used in alarm queries, see Alarm.

- **Statistic**: The aggregation function.

  ### Statistic values

  - **COUNT**- The number of observations received in the specified time period.
  - **MAX** - The highest value observed during the specified time period.

- **MEAN** - The value of Sum divided by Count during the specified time period.
- **MIN** - The lowest value observed during the specified time period.
- **P50** - The value of the 50th percentile.
- **P90** - The value of the 90th percentile.
- **P95** - The value of the 95th percentile.
- **P99** - The value of the 99th percentile.
- **P99.5** - The value of the 99.5th percentile.
- **RATE** - The per-interval average rate of change.
- **SUM** - All values added together.

- **Metric dimensions**: Optional filters to narrow the metric data evaluated.

  Dimension fields

  - **Dimension Name**: A qualifier specified in the metric definition. For example, the dimension `resourceId` is specified in the metric definition for `CpuUtilization`.

    Note

    Long lists of dimensions are trimmed.

    - To view dimensions by name, type one or more characters in the box. A refreshed (trimmed) list shows matching dimension names.
    - To retrieve all dimensions for a given metric, use the following API operation: ListMetrics

  - **Dimension Value**: The value you want to use for the specified dimension. For example, the resource identifier for your instance of interest.
  - **+ Additional dimension**: Adds another name-value pair for a dimension.

- **Trigger rule**: The condition that must be satisfied for the alarm to be in the firing state. The condition can specify a threshold, such as 90% for CPU Utilization, or an absence.

- **Operator**: The operator used in the condition threshold.

  Operator values

  - **greater than**
  - **greater than or equal to**
  - **equal to**
  - **less than**
  - **less than or equal to**
  - **between** (inclusive of specified values)
  - **outside** (inclusive of specified values)
  - **absent**

- **Value**: The value to use for the condition threshold.
- **Trigger Delay Minutes**: The number of minutes that the condition must be maintained before the alarm is in firing state.

Advanced Mode

Click **Advanced Mode** or **Switch to Advanced Mode** to view the alarm query as a Monitoring Query Language (MQL) expression. Edit your query using MQL syntax to aggregate results by group or for additional parameter values. See Monitoring Query Language (MQL) Reference.

- **Alarm Name**: User-friendly name for the new alarm.
- **Alarm Severity**: The perceived type of response required when the alarm is in the firing state.

- **Alarm Body**: The human-readable content of the notification delivered. Oracle recommends providing guidance to operators for resolving the alarm condition. Consider adding links to standard runbook practices. Example: "High CPU usage alert. Follow runbook instructions for resolution."
- **Tags (optional):** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
- **Metric description, dimensions, and trigger rule**: The metric to evaluate for the alarm condition, including dimensions and the trigger rule.
    - **Compartment**: The compartment containing the resources that emit the metrics evaluated by the alarm. The selected compartment is also the storage location of the alarm. By default, the first accessible compartment is selected.
    - **Metric Namespace**: The service or application emitting metrics for the resources that you want to monitor.
    - **Query Code Editor** box: The alarm query as a Monitoring Query Language (MQL) expression.

      Example alarm query:

      ```
      CpuUtilization[1m]{availabilityDomain=AD1}.groupBy(poolId).percentile(0.9) > 85
      ```

      For query syntax and examples, see Working with Metric Queries.
    - **Trigger Delay Minutes**: The number of minutes that the condition must be maintained before the alarm is in firing state.

The chart below the **Define alarm** section dynamically displays the last six hours of emitted metrics according to currently selected fields for the query. Very small or large values are indicated by International System of Units (SI units), such as M for mega (10 to the sixth power).

5. Under **Notifications**, select or create at least one notification destination:

Notifications settings

- **Destinations**:
  - **Destination Service**: The provider of the destination to use for notifications.
    Available options:
    - **Notifications Service**.
  - **Compartment**: The compartment storing the topic to be used for notifications. Can be a different compartment from the alarm and metric. By default, the first accessible compartment is selected.
  - **Topic**: The topic to use for notifications. Each topic supports a subscription protocol, such as PagerDuty.
  - **Create a topic**: Sets up a topic and subscription protocol in the selected compartment, using the specified destination service.
    - **Topic Name**: User-friendly name for the new topic. Example: "Operations Team " for a topic used to notify operations staff of firing alarms.
    - **Topic Description**: Description of the new topic.
    - **Subscription Protocol**: Medium of communication to use for the new topic. Configure your subscription for the protocol you want:

Email subscription

- **Subscription protocol**: Select **Email**.
- **Email Addresses**: Type an email address.

PagerDuty subscription

- **Subscription Protocol**: Select **HTTPS (PagerDuty)**.
- **PagerDuty Integration URL**: Type (or copy and paste) the *integration key* portion of the URL for your PagerDuty subscription. (The other portions of the URL are hard-coded.)
  For information on setting up and retrieving your integration key, see the PagerDuty documentation.
  Example integration key: Your68a4Integration4dd8Key12f4f6
  Example integration URL: https://events.pagerduty.com/integration/Your68a4Integration4dd8Key12f4f6/enqueue

◦ **+ Additional destination service**: Adds another destination service and topic to use for notifications.



Note

Each alarm is limited to one destination per supported destination service.

- **Repeat Notification?**: While the alarm is in the firing state, resends notifications at the specified interval.

- **Notification Interval**: The period of time to wait before resending the notification.
- **Suppress Notifications**: Sets up a suppression time window during which to suspend evaluations and notifications. Useful for avoiding alarm notifications during system maintenance periods.
  - **Suppression Description**
  - **Start Time**
  - **End Time**

6. If you want to disable the new alarm, clear **Enable This Alarm?**.
7. Click **Save alarm**.
   The new alarm is listed on the **Alarm Definitions** page.
   For more information about alarms, see Alarms Feature Overview.

resource page

Examples of resource pages are Compute instance detail pages and Block Volume volume detail pages. Alarms are available from these pages for resources that emit metrics.

To create an alarm from a chart query (resource page)

1. To view charts: On the resource page, under **Resources**, click **Metrics**.
2. At the top of the chart you're interested in, go to **Options**, and then select **Create an Alarm on this Query**.
3. On the **Create Alarm** page, under **Define alarm**, add the trigger and fill in or update other alarm settings as needed:

Alarm settings

Basic Mode (default)

By default, this page uses **Basic Mode**, which separates the metric from its dimensions and its trigger rule.

- **Alarm Name**: User-friendly name for the new alarm.
- **Alarm Severity**: The perceived type of response required when the alarm is in the firing state.
- **Alarm Body**: The human-readable content of the notification delivered. Oracle recommends providing guidance to operators for resolving the alarm condition. Consider adding links to standard runbook practices. Example: "High CPU usage alert. Follow runbook instructions for resolution."
- **Tags (optional):** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
- **Metric description**: The metric to evaluate for the alarm condition.
    - **Compartment**: The compartment containing the resources that emit the metrics evaluated by the alarm. The selected compartment is also the storage location of the alarm. By default, the first accessible compartment is selected.
    - **Metric Namespace**: The service or application emitting metrics for the resources that you want to monitor.
    - **Metric Name**: The name of the metric. Only one metric can be specified.

Example: **CpuUtilization**

- **Interval**: The aggregation window, or the frequency at which data points are aggregated.

  ### Interval values

  - **1m** - 1 minute
  - **5m** - 5 minutes
  - **1h** - 1 hour

  Note

  For alarm queries, the specified interval has no effect on the resolution of the request. The only valid value of the resolution for an alarm query request is `1m`. For more information about the resolution parameter as used in alarm queries, see Alarm.

- **Statistic**: The aggregation function.

  ### Statistic values

  - **COUNT**- The number of observations received in the specified time period.

- **MAX** - The highest value observed during the specified time period.
- **MEAN** - The value of Sum divided by Count during the specified time period.
- **MIN** - The lowest value observed during the specified time period.
- **P50** - The value of the 50th percentile.
- **P90** - The value of the 90th percentile.
- **P95** - The value of the 95th percentile.
- **P99** - The value of the 99th percentile.
- **P99.5** - The value of the 99.5th percentile.
- **RATE** - The per-interval average rate of change.
- **SUM** - All values added together.

- **Metric dimensions**: Optional filters to narrow the metric data evaluated.

  Dimension fields

  - **Dimension Name**: A qualifier specified in the metric definition. For example, the dimension `resourceId` is specified in the metric definition for `CpuUtilization`.

    Note

    Long lists of dimensions are trimmed.

    - To view dimensions by name, type one or more characters in the box. A refreshed (trimmed) list shows matching dimension names.
    - To retrieve all dimensions for a given metric, use the following API operation: ListMetrics

  - **Dimension Value**: The value you want to use for the specified dimension. For example, the resource identifier for your instance of interest.
  - **+ Additional dimension**: Adds another name-value pair for a dimension.

- **Trigger rule**: The condition that must be satisfied for the alarm to be in the firing state. The condition can specify a threshold, such as 90% for CPU Utilization, or an absence.

- **Operator**: The operator used in the condition threshold.

  Operator values

  - **greater than**
  - **greater than or equal to**
  - **equal to**
  - **less than**
  - **less than or equal to**
  - **between** (inclusive of specified values)
  - **outside** (inclusive of specified values)
  - **absent**

- **Value**: The value to use for the condition threshold.
- **Trigger Delay Minutes**: The number of minutes that the condition must be maintained before the alarm is in firing state.

Advanced Mode

Click **Advanced Mode** or **Switch to Advanced Mode** to view the alarm query as a Monitoring Query Language (MQL) expression. Edit your query using MQL syntax to aggregate results by group or for additional parameter values. See Monitoring Query Language (MQL) Reference.

- **Alarm Name**: User-friendly name for the new alarm.
- **Alarm Severity**: The perceived type of response required when the alarm is in the firing state.

- **Alarm Body**: The human-readable content of the notification delivered. Oracle recommends providing guidance to operators for resolving the alarm condition. Consider adding links to standard runbook practices. Example: "High CPU usage alert. Follow runbook instructions for resolution."
- **Tags (optional):** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
- **Metric description, dimensions, and trigger rule**: The metric to evaluate for the alarm condition, including dimensions and the trigger rule.
    - **Compartment**: The compartment containing the resources that emit the metrics evaluated by the alarm. The selected compartment is also the storage location of the alarm. By default, the first accessible compartment is selected.
    - **Metric Namespace**: The service or application emitting metrics for the resources that you want to monitor.
    - **Query Code Editor** box: The alarm query as a Monitoring Query Language (MQL) expression.

      Example alarm query:

      ```
      CpuUtilization[1m]{availabilityDomain=AD1}.groupBy(poolId).percentile(0.9) > 85
      ```

      For query syntax and examples, see Working with Metric Queries.
    - **Trigger Delay Minutes**: The number of minutes that the condition must be maintained before the alarm is in firing state.

The chart below the **Define alarm** section dynamically displays the last six hours of emitted metrics according to currently selected fields for the query. Very small or large values are indicated by International System of Units (SI units), such as M for mega (10 to the sixth power).

4. Under **Notifications**, select or create at least one notification destination:

Notifications settings

- **Destinations**:
  - **Destination Service**: The provider of the destination to use for notifications.
    Available options:
    - **Notifications Service**.
  - **Compartment**: The compartment storing the topic to be used for notifications. Can be a different compartment from the alarm and metric. By default, the first accessible compartment is selected.
  - **Topic**: The topic to use for notifications. Each topic supports a subscription protocol, such as PagerDuty.
  - **Create a topic**: Sets up a topic and subscription protocol in the selected compartment, using the specified destination service.
    - **Topic Name**: User-friendly name for the new topic. Example: "Operations Team " for a topic used to notify operations staff of firing alarms.
    - **Topic Description**: Description of the new topic.
    - **Subscription Protocol**: Medium of communication to use for the new topic. Configure your subscription for the protocol you want:

Email subscription

- **Subscription protocol**: Select **Email**.
- **Email Addresses**: Type an email address.

PagerDuty subscription

- **Subscription Protocol**: Select **HTTPS (PagerDuty)**.
- **PagerDuty Integration URL**: Type (or copy and paste) the *integration key* portion of the URL for your PagerDuty subscription. (The other portions of the URL are hard-coded.)
  For information on setting up and retrieving your integration key, see the PagerDuty documentation.
  Example integration key: Your68a4Integration4dd8Key12f4f6
  Example integration URL: https://events.pagerduty.com/integration/Your68a4Integration4dd8Key12f4f6/enqueue

◦ **+ Additional destination service**: Adds another destination service and topic to use for notifications.

> **Note**
>
> Each alarm is limited to one destination per supported destination service.

- **Repeat Notification?**: While the alarm is in the firing state, resends notifications at the specified interval.

- **Notification Interval**: The period of time to wait before resending the notification.
- **Suppress Notifications**: Sets up a suppression time window during which to suspend evaluations and notifications. Useful for avoiding alarm notifications during system maintenance periods.
  - **Suppression Description**
  - **Start Time**
  - **End Time**

5. If you want to disable the new alarm, clear **Enable This Alarm?**.
6. Click **Save alarm**.
   The new alarm is listed on the **Alarm Definitions** page.
   For more information about alarms, see Alarms Feature Overview.

Building Metric Queries

This topic describes how to query metrics for resources of interest, create alarms from a given query, and share Console charts.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Prerequisites

- IAM policies: Querying metrics is part of monitoring. To monitor resources, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. The policy must give you access to the monitoring services as well as the resources being monitored. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in. For more information on user authorizations for monitoring, see the Authentication and Authorization section for the related service: Monitoring or Notifications.
- Metrics exist in Monitoring: The resources that you want to monitor must emit metrics to the Monitoring service.
- Compute instances: To emit metrics, Compute instances must be monitoring-enabled. OracleCloudAgent software installation may also be required. For more information, see Enabling Monitoring for Compute Instances.

Working with Metric Queries

This section shows MQL syntax of metric and alarm queries.

Use metric queries to actively and passively monitor your cloud resources. Actively monitor with metric queries generated on the fly. In the Console, update a chart to show data from multiple queries. Make sure that you store queries you want to reuse. Passively monitor with alarms that add a condition, or trigger rule, to a metric query.

Metric query syntax (boldface elements are required):

`metric[interval]``{dimensionname=dimensionvalue}.groupingfunction.`**`statistic`**

Threshold Alarm query syntax (boldface elements are required):

`metric[interval]``{dimensionname=dimensionvalue}.groupingfunction.`**`statistic alarmoperator alarmvalue`**

For supported parameter values, see Monitoring Query Language (MQL) Reference.

Example queries

Simple metric query

Maximum CPU Utilization at a one-minute interval.

Number of lines displayed in the metric chart (Console): 1 per resource.

```
CpuUtilization[1m].max()
```

Filtered metric query

Maximum CPU Utilization at a one-minute interval, filtered to a single resource.

Number of lines displayed in the metric chart (Console): 1.

```
CpuUtilization[1m]{resourceId="ocid1.instance.oc1.phx.exampleuniqueID"}.max()
```

Aggregated metric query

All IopsRead at a one-minute interval, filtered to a compartment, aggregated for the maximum.

Number of lines displayed in the metric chart (Console): 1.

```
IopsRead[1m]
{compartmentId="ocid1.compartment.oc1.phx..exampleuniqueID"}.grouping().max()
```

Group-aggregated metric query

Aggregated average of CPU Utilization by availability domain and pool ID, filtered to Compute instances that use the specified shape.

Number of lines displayed in the metric chart (Console): 1 per pool and 1 per availability domain.

```
CPUUtilization[1m]{shape="VM.Standard2.8"}.groupBy
(availabilityDomain,poolId).mean()
```

Alarm query (threshold)

Triggered when the 90th percentile of CPU Utilization, aggregated by pool ID and filtered to the specified availability domain, exceeds 85.

Number of lines displayed in the metric chart (Console): 1 per pool.

```
CpuUtilization[1m]{availabilityDomain="VeBZ:PHX-AD-1"}.groupBy
(poolId).percentile(0.9) > 85
```

For background information on metrics in Oracle Cloud Infrastructure, see Metrics Feature Overview.

Using the Console

To create a query

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Metrics Explorer**.
   The **Metrics Explorer** page displays an empty chart with fields to build a query.
2. Fill in the fields for a new query.
   - **Compartment**: The compartment containing the resources that you want to monitor. By default, the first accessible compartment is selected.
   - **Metric Namespace**: The service or application emitting metrics for the resources that you want to monitor.
   - **Metric Name**: The name of the metric. Only one metric can be specified. Metric selections depend on the selected compartment and metric namespace.
     Example: **CpuUtilization**
   - **Interval**: The aggregation window.

Interval values

- **1m** - 1 minute
- **5m** - 5 minutes
- **1h** - 1 hour



Note

For metric queries, the interval you select drives the default resolution of the request, which determines the maximum time range of data returned.

Maximum time range returned for a query

The maximum time range returned for a metric query depends on the resolution. By default, for metric queries, the resolution is the same as the query interval. Following are the maximum time ranges returned for each interval selection available in the Console.

Interval	Default resolution (metric queries)	Maximum time range returned
1h	1 hour	90 days
5m	5 minutes	30 days
1m	1 minute	7 days



For more information about the resolution parameter as used in metric queries, see SummarizeMetricsData.

- **Statistic**: The aggregation function.

  Statistic values

  - **COUNT**- The number of observations received in the specified time period.
  - **MAX** - The highest value observed during the specified time period.
  - **MEAN** - The value of Sum divided by Count during the specified time period.
  - **MIN** - The lowest value observed during the specified time period.
  - **P50** - The value of the 50th percentile.
  - **P90** - The value of the 90th percentile.
  - **P95** - The value of the 95th percentile.
  - **P99** - The value of the 99th percentile.
  - **P99.5** - The value of the 99.5th percentile.
  - **RATE** - The per-interval average rate of change.
  - **SUM** - All values added together.

- **Metric dimensions**: Optional filters to narrow the metric data evaluated.

  Dimension fields

  - **Dimension Name**: A qualifier specified in the metric definition. For example, the dimension `resourceId` is specified in the metric definition for `CpuUtilization`.

    Note

    Long lists of dimensions are trimmed.

    - To view dimensions by name, type one or more characters in the box. A refreshed (trimmed) list shows matching dimension names.
    - To retrieve all dimensions for a given metric, use the following API operation: ListMetrics

  - **Dimension Value**: The value you want to use for the specified dimension. For example, the resource identifier for your instance of interest.
  - **+ Additional dimension**: Adds another name-value pair for a dimension.

- **Aggregate Metric Streams**: Aggregates all results to plot a single aggregated average for all metric streams. This average is plotted as a single line on the metric chart. This operation is helpful when you want to plot a metric as one line for all resources.

3. Click **Update Chart**.
   The chart shows the results of your new query. Very small or large values are indicated by International System of Units (SI units), such as M for mega (10 to the sixth power). Units correspond to the selected metric and do not change by statistic.

   ### Troubleshooting Errors and Query Limits

   If you see an error that the query has exceeded the maximum number of metric streams, then update the query to evaluate a number of metric streams that is within the limit. For example, you can reduce the metric streams by specifying dimensions. You can continue to evaluate all metric streams that were in the original query by spreading the metric streams across multiple queries (or alarms).
   Limits information for returned data includes the 100,000 data point maximum and time range maximums (determined by resolution, which relates to interval). See MetricData Reference.

4. To customize the y-axis label or range, type the label you want into **Y-Axis Label** or type the minimum and maximum values you want into **Y-Axis Min Value** and **Y-Axis Max Value**.
   Only numeric characters are allowed for custom ranges. Custom labels and ranges are not persisted in shared queries (MQL).
5. To view the query as a Monitoring Query Language (MQL) expression, click **Advanced Mode**.
   Advanced Mode is located on the right, under the chart.
   Use Advanced Mode to edit your query using MQL syntax to aggregate results by group. The MQL syntax also supports additional parameter values. For more information about query parameters in Basic Mode and Advanced Mode, see Monitoring Query Language (MQL) Reference.
6. To create another query, click **Add Query** below the chart.

To change the time range

1. View the **Metrics Explorer** page: Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Metrics Explorer**.
2. To select a period of time, such as **Last hour**, click **Start Time** or **End Time**.
3. To enter a time value, click in **Start Time** or **End Time** and then type a value.

To filter results

Filter results to limit the data plotted on the metric chart. For example, filter results to a resource or pool of interest.

Filtering is done through selected dimensions; available dimensions vary by metric.

1. View the **Metrics Explorer** page: Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Metrics Explorer**.
2. If necessary, open the query for editing: Click the **Edit query** icon.
3. Under **Metric dimensions**, select a **Dimension Name** and **Dimension Value**.

Dimension fields

- **Dimension Name**: A qualifier specified in the metric definition. For example, the dimension `resourceId` is specified in the metric definition for `CpuUtilization`.



Note

Long lists of dimensions are trimmed.

- To view dimensions by name, type one or more characters in the box. A refreshed (trimmed) list shows matching dimension names.
- To retrieve all dimensions for a given metric, use the following API operation: ListMetrics

- **Dimension Value**: The value you want to use for the specified dimension. For example, the resource identifier for your instance of interest.
- **+ Additional dimension**: Adds another name-value pair for a dimension.

4. To add a dimension name-value pair, click **+ Additional dimension**.
5. Click **Update Chart**.
   The chart shows the filtered results of your query.

Troubleshooting Errors and Query Limits

If you see an error that the query has exceeded the maximum number of metric streams, then update the query to evaluate a number of metric streams that is within the limit. For example, you can reduce the metric streams by specifying dimensions. You can continue to evaluate all metric streams that were in the original query by spreading the metric streams across multiple queries (or alarms).

Limits information for returned data includes the 100,000 data point maximum and time range maximums (determined by resolution, which relates to interval). See MetricData Reference.

6. To view the query as a Monitoring Query Language (MQL) expression, click **Advanced Mode**.
   The dimension name-value fragment appears after the metric-interval fragment.
   In the following example query, the dimension name-value fragment is `{resourceId="ocid1.instance.oc1.phx.exampleuniqueID"}`, which filters results by the specified resource identifier.

   ```
   CpuUtilization[1m]
   {resourceId="ocid1.instance.oc1.phx.exampleuniqueID"}.max()
   ```

   The MQL syntax supports additional parameter values. See Monitoring Query Language (MQL) Reference.

To aggregate all results

Aggregate all results to plot a single aggregated average for all metric streams. This average is plotted as a single line on the metric chart. This operation is helpful when you want to plot a metric as one line for all resources.

1. View the **Metrics Explorer** page: Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Metrics Explorer**.
2. If necessary, open the query for editing: Click the **Edit query** icon.
3. Click **Aggregate Metric Streams**.

4. Click **Update Chart**.
   The chart shows the results of your query.

Troubleshooting Errors and Query Limits

If you see an error that the query has exceeded the maximum number of metric streams, then update the query to evaluate a number of metric streams that is within the limit. For example, you can reduce the metric streams by specifying dimensions. You can continue to evaluate all metric streams that were in the original query by spreading the metric streams across multiple queries (or alarms).

Limits information for returned data includes the 100,000 data point maximum and time range maximums (determined by resolution, which relates to interval). See MetricData Reference.

5. To view the query as a Monitoring Query Language (MQL) expression, click **Advanced Mode**.
   The `grouping()` function appears before the statistic. For example, the following query returns the maximum (`max()`) IopsRead metric data at a one-minute interval, filtered to a compartment, with all results aggregated.

```
IopsRead[1m]{compartmentID = "<compartment_OCID>"}.grouping().max()
```

   Edit your query in MQL to aggregate results by group. The MQL syntax also supports additional parameter values. See Monitoring Query Language (MQL) Reference.

To aggregate results by group



Note

Aggregating query results by group requires the `groupBy()` function, which is available in **Advanced Mode** only.

Aggregate query results by group to plot group-specific aggregated averages. Each group's average is plotted as a single line on the metric chart. This operation is helpful when you want to identify trends by group rather than individual resource.

For example, the following query returns the average (`mean()`) CPU Utilization metric data at a one-minute interval. Results are filtered to the specified shape and grouped by availability domain. If you have Compute instances of this shape sending metrics across three availability domains, then three lines are plotted on the metric chart.

```
CPUUtilization[1m]{shape="VM.Standard1.1"}.groupBy(availabilityDomain).mean()
```

1. View the **Metrics Explorer** page: Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Metrics Explorer**.
2. If necessary, open the query for editing: Click the **Edit query** icon.
3. Select **Advanced Mode** below the chart on the right.
4. In the **Query Code Editor** box, insert the `groupBy({dimension})` function between the metric-interval fragment and the statistic, where {dimension} is the name of a dimension provided in the definition of the indicated metric.

   For example, insert the following fragment to group by availability domain, assuming that the dimension is available for the selected metric.

   ```
   groupBy(availabilityDomain)
   ```

   The MQL syntax supports additional parameter values. See Monitoring Query Language (MQL) Reference.

5. Click **Update Chart**.
   The chart is updated to show a single line for each grouped result.

Troubleshooting Errors and Query Limits

If you see an error that the query has exceeded the maximum number of metric streams, then update the query to evaluate a number of metric streams that is within the limit. For example, you can reduce the metric streams by specifying dimensions. You can continue to evaluate all metric streams that were in the original query by spreading the metric streams across multiple queries (or alarms).

Limits information for returned data includes the 100,000 data point maximum and time range maximums (determined by resolution, which relates to interval). See MetricData Reference.

To edit a query using MQL syntax

Edit your query using MQL syntax to aggregate results by group or for additional parameter values. See Monitoring Query Language (MQL) Reference.

1. View the **Metrics Explorer** page: Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Metrics Explorer**.
2. If necessary, open the query for editing: Click the **Edit Query** icon.
3. Click **Advanced Mode**.
   The query is displayed as a Monitoring Query Language (MQL) expression.
4. In the **Query Code Editor** box, edit the query as needed.

5. Click **Update Chart**.
   The chart is updated.

Troubleshooting Errors and Query Limits

If you see an error that the query has exceeded the maximum number of metric streams, then update the query to evaluate a number of metric streams that is within the limit. For example, you can reduce the metric streams by specifying dimensions. You can continue to evaluate all metric streams that were in the original query by spreading the metric streams across multiple queries (or alarms).

Limits information for returned data includes the 100,000 data point maximum and time range maximums (determined by resolution, which relates to interval). See MetricData Reference.

To create an alarm from a query

Create an alarm to passively monitor for a condition in results from metric queries. Creating an alarm from a query involves adding a trigger rule to the query and setting up notifications.

1. View the **Metrics Explorer** page: Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Metrics Explorer**.
2. If necessary, open the query for editing: Click the **Edit Query** icon.
3. Click **Create Alarm**.
4. On the **Create Alarm** page, under **Define alarm**, add the trigger and fill in or update other alarm settings as needed:

Alarm settings

Basic Mode (default)

By default, this page uses **Basic Mode**, which separates the metric from its dimensions

and its trigger rule.

- **Alarm Name**: User-friendly name for the new alarm.
- **Alarm Severity**: The perceived type of response required when the alarm is in the firing state.
- **Alarm Body**: The human-readable content of the notification delivered. Oracle recommends providing guidance to operators for resolving the alarm condition. Consider adding links to standard runbook practices. Example: "High CPU usage alert. Follow runbook instructions for resolution."
- **Tags (optional):** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
- **Metric description**: The metric to evaluate for the alarm condition.
    - **Compartment**: The compartment containing the resources that emit the metrics evaluated by the alarm. The selected compartment is also the storage location of the alarm. By default, the first accessible compartment is selected.
    - **Metric Namespace**: The service or application emitting metrics for the resources that you want to monitor.
    - **Metric Name**: The name of the metric. Only one metric can be specified. Example: **CpuUtilization**

- **Interval**: The aggregation window, or the frequency at which data points are aggregated.

  ### Interval values

  - **1m** - 1 minute
  - **5m** - 5 minutes
  - **1h** - 1 hour

  > **Note**
  >
  > For alarm queries, the specified interval has no effect on the resolution of the request. The only valid value of the resolution for an alarm query request is `1m`. For more information about the resolution parameter as used in alarm queries, see Alarm.

- **Statistic**: The aggregation function.

  ### Statistic values

  - **COUNT**- The number of observations received in the specified time period.
  - **MAX** - The highest value observed during the specified time period.
  - **MEAN** - The value of Sum divided by Count during the specified time period.

- **MIN** - The lowest value observed during the specified time period.
- **P50** - The value of the 50th percentile.
- **P90** - The value of the 90th percentile.
- **P95** - The value of the 95th percentile.
- **P99** - The value of the 99th percentile.
- **P99.5** - The value of the 99.5th percentile.
- **RATE** - The per-interval average rate of change.
- **SUM** - All values added together.

- **Metric dimensions**: Optional filters to narrow the metric data evaluated.

  Dimension fields

  - **Dimension Name**: A qualifier specified in the metric definition. For example, the dimension `resourceId` is specified in the metric definition for `CpuUtilization`.

    Note

    Long lists of dimensions are trimmed.

    - To view dimensions by name, type one or more characters in the box. A refreshed (trimmed) list shows matching dimension names.
    - To retrieve all dimensions for a given metric, use the following API operation: ListMetrics

  - **Dimension Value**: The value you want to use for the specified dimension. For example, the resource identifier for your instance of interest.
  - **+ Additional dimension**: Adds another name-value pair for a dimension.

- **Trigger rule**: The condition that must be satisfied for the alarm to be in the firing state. The condition can specify a threshold, such as 90% for CPU Utilization, or an absence.

- **Operator**: The operator used in the condition threshold.

  Operator values

  - **greater than**
  - **greater than or equal to**
  - **equal to**
  - **less than**
  - **less than or equal to**
  - **between** (inclusive of specified values)
  - **outside** (inclusive of specified values)
  - **absent**

- **Value**: The value to use for the condition threshold.
- **Trigger Delay Minutes**: The number of minutes that the condition must be maintained before the alarm is in firing state.

Advanced Mode

Click **Advanced Mode** or **Switch to Advanced Mode** to view the alarm query as a Monitoring Query Language (MQL) expression. Edit your query using MQL syntax to aggregate results by group or for additional parameter values. See Monitoring Query Language (MQL) Reference.

- **Alarm Name**: User-friendly name for the new alarm.
- **Alarm Severity**: The perceived type of response required when the alarm is in the firing state.
- **Alarm Body**: The human-readable content of the notification delivered. Oracle recommends providing guidance to operators for resolving the alarm condition.

Consider adding links to standard runbook practices. Example: "High CPU usage alert. Follow runbook instructions for resolution."

- **Tags (optional):** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
- **Metric description, dimensions, and trigger rule**: The metric to evaluate for the alarm condition, including dimensions and the trigger rule.
  - **Compartment**: The compartment containing the resources that emit the metrics evaluated by the alarm. The selected compartment is also the storage location of the alarm. By default, the first accessible compartment is selected.
  - **Metric Namespace**: The service or application emitting metrics for the resources that you want to monitor.
  - **Query Code Editor** box: The alarm query as a Monitoring Query Language (MQL) expression.
    Example alarm query:

    ```
    CpuUtilization[1m]{availabilityDomain=AD1}.groupBy(poolId).percentile(0.9) > 85
    ```

    For query syntax and examples, see Working with Metric Queries.
  - **Trigger Delay Minutes**: The number of minutes that the condition must be maintained before the alarm is in firing state.

The chart below the **Define alarm** section dynamically displays the last six hours of emitted metrics according to currently selected fields for the query. Very small or large values are indicated by International System of Units (SI units), such as M for mega (10 to the sixth power).

5. Under **Notifications**, select or create at least one notification destination:

Notifications settings

- **Destinations**:
  - **Destination Service**: The provider of the destination to use for notifications.
    Available options:
    - **Notifications Service**.
  - **Compartment**: The compartment storing the topic to be used for notifications. Can be a different compartment from the alarm and metric. By default, the first accessible compartment is selected.
  - **Topic**: The topic to use for notifications. Each topic supports a subscription protocol, such as PagerDuty.
  - **Create a topic**: Sets up a topic and subscription protocol in the selected compartment, using the specified destination service.
    - **Topic Name**: User-friendly name for the new topic. Example: "Operations Team " for a topic used to notify operations staff of firing alarms.
    - **Topic Description**: Description of the new topic.
    - **Subscription Protocol**: Medium of communication to use for the new topic. Configure your subscription for the protocol you want:

      Email subscription
      - **Subscription protocol**: Select **Email**.
      - **Email Addresses**: Type an email address.

PagerDuty subscription

- **Subscription Protocol**: Select **HTTPS (PagerDuty)**.
- **PagerDuty Integration URL**: Type (or copy and paste) the *integration key* portion of the URL for your PagerDuty subscription. (The other portions of the URL are hard-coded.)
  For information on setting up and retrieving your integration key, see the PagerDuty documentation.
  Example integration key: Your68a4Integration4dd8Key12f4f6
  Example integration URL: https://events.pagerduty.com/integration/Your68a4Integration4dd8Key12f4f6/enqueue

- **+ Additional destination service**: Adds another destination service and topic to use for notifications.



Note

Each alarm is limited to one destination per supported destination service.

- **Repeat Notification?**: While the alarm is in the firing state, resends notifications at the specified interval.
- **Notification Interval**: The period of time to wait before resending the notification.

- **Suppress Notifications**: Sets up a suppression time window during which to suspend evaluations and notifications. Useful for avoiding alarm notifications during system maintenance periods.
  - **Suppression Description**
  - **Start Time**
  - **End Time**

6. If you want to disable the new alarm, clear **Enable This Alarm?**.
7. Click **Save alarm**.
   The new alarm is listed on the **Alarm Definitions** page.
   For more information about alarms, see Alarms Feature Overview.

To hide a query from the chart

1. View the **Metrics Explorer** page: Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Metrics Explorer**.
2. Click the **Toggle query on chart** icon for the query that you want to hide.

To share a query

1. View the **Metrics Explorer** page: Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Metrics Explorer**.
2. If necessary, open the query for editing: Click the **Edit query** icon.
3. Click **Advanced Mode**.
4. In the **Query Code Editor** box, copy the query.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line

Interface.

Use this API operation to find metric names and dimensions (view metric definitions):

ListMetrics

Use this API operation to query metrics by name (and optionally filter by dimension):

SummarizeMetricsData

Publishing Custom Metrics

This topic describes how to publish your own custom metrics to the Monitoring service.

You can publish your own metrics to Monitoring using the API. You can view charts of your published metrics using the Console , query metrics using the API, and set up alarms using the Console or API.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Prerequisites

IAM policies: To publish custom metrics, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in. Administrators: For a related common policy, see Let users publish custom metrics.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.



Note

When defining your custom metrics, note the following:

- Make sure your custom metrics do not exceed limits. For example, note the valid range of dimensions and maximum number of streams for custom metrics. See PostMetricData.
- Define your metrics with aggregation in mind. While custom metrics can be posted as frequently as every second (minimum frequency of one second), the minimum aggregation interval is one minute.
- Define your metrics with return limits in mind. Limits information for returned data includes the 100,000 data point maximum and time range maximums (determined by resolution, which relates to interval). See MetricData Reference.

Use this API operation to publish custom metrics:

PostMetricData



Note

Oracle recommends the following:

- Send batched requests to maximize metric streams per request. A batched request contains multiple metrics or metric namespaces. Note limits. See PostMetricData.
- Publish metrics only when relevant contexts require monitoring; that is, when data points need to be collected. If you want to publish metrics during inactive periods when no observations exist, then you can manually create "0" values for publishing.

Example of a batched request

This example shows a single request containing data points for metrics across two metric namespaces.

```
[
   {
      "namespace":"myfirstnamespace",
      "compartmentId":"ocid1.compartment.oc1..exampleuniqueID",
      "name":"successRate",
      "dimensions":{
         "resourceId":"ocid1.exampleresource.region1.phx.exampleuniqueID",
         "appName":"myAppA"
      },
      "metadata":{
         "unit":"percent",
         "displayName":"MyAppA Success Rate"
      },
      "datapoints":[
         {
```

```
            "timestamp":"2019-03-10T22:19:20Z",
            "value":83.0
         },
         {
            "timestamp":"2019-03-10T22:19:40Z",
            "value":90.1
         }
      ]
   },
   {
      "namespace":"mysecondnamespace",
      "compartmentId":"ocid1.compartment.oc1..exampleuniqueID",
      "name":"deliveryRate",
      "dimensions":{
         "resourceId":"ocid1.exampleresource.region1.phx.exampleuniqueID",
         "appName":"myAppB"
      },
      "metadata":{
         "unit":"bytes",
         "displayName":"MyAppB Delivery Rate"
      },
      "datapoints":[
         {
            "timestamp":"2019-03-10T22:19:00Z",
            "value":87.0,
            "count":60
         },
         {
            "timestamp":"2019-03-10T22:19:00Z",
            "value":96.0,
            "count":30
         }
      ]
   }
]
```

You can access your published custom metrics the same way you access any other metrics stored by the Monitoring service. View charts from queries using the Console, query metrics using the CLI or API, and set up alarms using the Console, CLI, or API.

Managing Alarms

This topic describes how to create, update, suppress, and delete alarms, as well as how to retrieve alarm history. See also Best Practices for your Alarms.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Prerequisites

- IAM policies: Managing alarms is part of monitoring. To monitor resources, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. The policy must give you access to the monitoring services as well as the resources being monitored. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in. For more information on user authorizations for monitoring, see the Authentication and Authorization section for the related service: Monitoring or Notifications.
- Metrics exist in Monitoring: The resources that you want to monitor must emit metrics to the Monitoring service.
- Compute instances: To emit metrics, Compute instances must be monitoring-enabled. OracleCloudAgent software installation may also be required. For more information, see Enabling Monitoring for Compute Instances.

Tagging Resources

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

Using the Console

To see all firing alarms

Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Alarm Status**.

You can suppress alarms during a given time range. You can also disable and delete alarms.

To create an alarm

This section includes steps to create example alarms as well as any kind of alarm.

To create an example threshold alarm

This procedure walks through creation of an example threshold alarm to detect Compute instances operating at non-optimal thresholds. A *threshold alarm* is an alarm that checks for metric values outside a given range or value. The procedure uses options as displayed in Basic Mode.

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Alarm Definitions**.
2. Click **Create alarm**.
3. On the **Create Alarm** page, under **Define alarm**, fill in or update the alarm settings:
    - **Alarm Name**: Non-Optimal Alarm
    - **Alarm Severity**: Warning

- **Alarm Body**: Non-optimal utilization detected. An application or process may be consuming more CPU than usual.
- **Metric description**:
  - **Compartment**: (select your compartment)
  - **Metric Namespace**: **oci_computeagent**
  - **Metric Name**: **CpuUtilization**
  - **Interval**: **1m**
  - **Statistic**: **Count**
- **Trigger rule**:
  - **Operator**: **between**
  - **Value**: 60
  - **Value**: 80
  - **Trigger Delay Minutes**: 10

4. Set up an email notification under **Notifications**, **Destinations**:
   - **Destination Service**: **Notifications Service**
   - **Compartment**: (select your compartment)
   - **Topic**: Click **Create a topic**
     - **Topic Name**: Operations Team
     - **Topic Description**: Resource Monitoring Channel
     - **Subscription Protocol**: **Email**
     - **Email Addresses**: (type an email address for the operations team here)
5. Repeat notifications every day:
   - **Repeat Notification?**: (select this option)
   - **Notification Interval**: 24 hours
6. Click **Save alarm**.

To create an example absence alarm

This procedure walks through creation of an example absence alarm to detect resources that may be down or unreachable. An *absence alarm* is an alarm that checks for absent metrics (using the absent operator). The procedure uses options as displayed in Basic Mode.

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Alarm Definitions**.
2. Click **Create alarm**.
3. On the **Create Alarm** page, under **Define alarm**, fill in or update the alarm settings:
    - **Alarm Name**: Up/Down Resource Alarm
    - **Alarm Severity**: Critical
    - **Alarm Body**: Resource may be down. Please investigate. Move workloads to another available resource.
    - **Metric description**:
        - **Compartment**: (select your compartment)
        - **Metric Namespace**: **oci_computeagent**
        - **Metric Name**: **CpuUtilization**
        - **Interval**: **1m**
        - **Statistic**: **Count**
    - **Trigger rule**:
        - **Operator**: **absent**
        - **Trigger Delay Minutes**: 5
4. Set up an email notification under **Notifications**, **Destinations**:
    - **Destination Service**: **Notifications Service**
    - **Compartment**: (select your compartment)
    - **Topic**: Click **Create a topic**

- **Topic Name**: Operations Team
- **Topic Description**: Resource Up/Down Channel
- **Subscription Protocol**: **Email**
- **Email Addresses**: (type an email address for the operations team here)



Note

To add PagerDuty notifications, create a copy of this alarm and choose the **HTTPS (PagerDuty)** protocol.

About the PagerDuty integration URL

Type (or copy and paste) the *integration key* portion of the URL for your PagerDuty subscription. (The other portions of the URL are hard-coded.)

For information on setting up and retrieving your integration key, see the PagerDuty documentation.

Example integration key: Your68a4Integration4dd8Key12f4f6

Example integration URL: https://events.pagerduty.com/integration/Your68a4Integration4dd8Key12f4f6/enqueue

5. Repeat notifications every minute:

- **Repeat Notification?**: (select this option)
- **Notification Interval**: 1 minute

6. Click **Save alarm**.

To create an alarm (any kind)

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Alarm Definitions**.
2. Click **Create alarm**.

> **Note**
>
> You can also create an alarm from a predefined query on the **Service Metrics** page. Expand **Options** and click **Create an Alarm on this Query**. For more information about service metrics, see Viewing Default Metric Charts.

3. On the **Create Alarm** page, under **Define alarm**, fill in or update the alarm settings:

> **Note**
>
> To toggle between Basic Mode and Advanced Mode, click **Switch to Advanced Mode** or **Switch to Basic Mode** (to the right of **Define Alarm**).

Basic Mode (default)

By default, this page uses **Basic Mode**, which separates the metric from its dimensions

and its trigger rule.

- **Alarm Name**: User-friendly name for the new alarm.
- **Alarm Severity**: The perceived type of response required when the alarm is in the firing state.
- **Alarm Body**: The human-readable content of the notification delivered. Oracle recommends providing guidance to operators for resolving the alarm condition. Consider adding links to standard runbook practices. Example: "High CPU usage alert. Follow runbook instructions for resolution."
- **Tags (optional):** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
- **Metric description**: The metric to evaluate for the alarm condition.
    - **Compartment**: The compartment containing the resources that emit the metrics evaluated by the alarm. The selected compartment is also the storage location of the alarm. By default, the first accessible compartment is selected.
    - **Metric Namespace**: The service or application emitting metrics for the resources that you want to monitor.
    - **Metric Name**: The name of the metric. Only one metric can be specified. Example: **CpuUtilization**

- **Interval**: The aggregation window, or the frequency at which data points are aggregated.

  ### Interval values

  - **1m** - 1 minute
  - **5m** - 5 minutes
  - **1h** - 1 hour

  Note

  For alarm queries, the specified interval has no effect on the resolution of the request. The only valid value of the resolution for an alarm query request is `1m`. For more information about the resolution parameter as used in alarm queries, see Alarm.

- **Statistic**: The aggregation function.

  ### Statistic values

  - **COUNT**- The number of observations received in the specified time period.
  - **MAX** - The highest value observed during the specified time period.
  - **MEAN** - The value of Sum divided by Count during the specified time period.

- **MIN** - The lowest value observed during the specified time period.
- **P50** - The value of the 50th percentile.
- **P90** - The value of the 90th percentile.
- **P95** - The value of the 95th percentile.
- **P99** - The value of the 99th percentile.
- **P99.5** - The value of the 99.5th percentile.
- **RATE** - The per-interval average rate of change.
- **SUM** - All values added together.

- **Metric dimensions**: Optional filters to narrow the metric data evaluated.

  Dimension fields

  - **Dimension Name**: A qualifier specified in the metric definition. For example, the dimension `resourceId` is specified in the metric definition for `CpuUtilization`.

    Note

    Long lists of dimensions are trimmed.

    - To view dimensions by name, type one or more characters in the box. A refreshed (trimmed) list shows matching dimension names.
    - To retrieve all dimensions for a given metric, use the following API operation: ListMetrics

  - **Dimension Value**: The value you want to use for the specified dimension. For example, the resource identifier for your instance of interest.
  - **+ Additional dimension**: Adds another name-value pair for a dimension.

- **Trigger rule**: The condition that must be satisfied for the alarm to be in the firing state. The condition can specify a threshold, such as 90% for CPU Utilization, or an absence.

- **Operator**: The operator used in the condition threshold.

  Operator values

  - **greater than**
  - **greater than or equal to**
  - **equal to**
  - **less than**
  - **less than or equal to**
  - **between** (inclusive of specified values)
  - **outside** (inclusive of specified values)
  - **absent**

- **Value**: The value to use for the condition threshold.
- **Trigger Delay Minutes**: The number of minutes that the condition must be maintained before the alarm is in firing state.

Advanced Mode

Click **Advanced Mode** or **Switch to Advanced Mode** to view the alarm query as a Monitoring Query Language (MQL) expression. Edit your query using MQL syntax to aggregate results by group or for additional parameter values. See Monitoring Query Language (MQL) Reference.

- **Alarm Name**: User-friendly name for the new alarm.
- **Alarm Severity**: The perceived type of response required when the alarm is in the firing state.
- **Alarm Body**: The human-readable content of the notification delivered. Oracle recommends providing guidance to operators for resolving the alarm condition.

Consider adding links to standard runbook practices. Example: "High CPU usage alert. Follow runbook instructions for resolution."

- **Tags (optional):** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
- **Metric description, dimensions, and trigger rule**: The metric to evaluate for the alarm condition, including dimensions and the trigger rule.
  - **Compartment**: The compartment containing the resources that emit the metrics evaluated by the alarm. The selected compartment is also the storage location of the alarm. By default, the first accessible compartment is selected.
  - **Metric Namespace**: The service or application emitting metrics for the resources that you want to monitor.
  - **Query Code Editor** box: The alarm query as a Monitoring Query Language (MQL) expression.
    Example alarm query:

    ```
    CpuUtilization[1m]{availabilityDomain=AD1}.groupBy(poolId).percentile(0.9) > 85
    ```

    For query syntax and examples, see Working with Metric Queries.
  - **Trigger Delay Minutes**: The number of minutes that the condition must be maintained before the alarm is in firing state.

The chart below the **Define alarm** section dynamically displays the last six hours of emitted metrics according to currently selected fields for the query. Very small or large values are indicated by International System of Units (SI units), such as M for mega (10 to the sixth power).

4. Set up notifications: Under **Notifications**, fill in the fields.

- **Destinations**:
  - **Destination Service**: The provider of the destination to use for notifications.

    Available options:
    - **Notifications Service**.
  - **Compartment**: The compartment storing the topic to be used for notifications. Can be a different compartment from the alarm and metric. By default, the first accessible compartment is selected.
  - **Topic**: The topic to use for notifications. Each topic supports a subscription protocol, such as PagerDuty.
  - **Create a topic**: Sets up a topic and subscription protocol in the selected compartment, using the specified destination service.
    - **Topic Name**: User-friendly name for the new topic. Example: "Operations Team " for a topic used to notify operations staff of firing alarms.
    - **Topic Description**: Description of the new topic.
    - **Subscription Protocol**: Medium of communication to use for the new topic. Configure your subscription for the protocol you want:

      Email subscription
      - **Subscription protocol**: Select **Email**.
      - **Email Addresses**: Type an email address.

      PagerDuty subscription
      - **Subscription Protocol**: Select **HTTPS (PagerDuty)**.

- **PagerDuty Integration URL**: Type (or copy and paste) the *integration key* portion of the URL for your PagerDuty subscription. (The other portions of the URL are hard-coded.)
  For information on setting up and retrieving your integration key, see the PagerDuty documentation.
  Example integration key: Your68a4Integration4dd8Key12f4f6
  Example integration URL: https://events.pagerduty.com/integration/Your68a4Integration4dd8Key12f4f6/enqueue

- **+ Additional destination service**: Adds another destination service and topic to use for notifications.



Note

Each alarm is limited to one destination per supported destination service.

- **Repeat Notification?**: While the alarm is in the firing state, resends notifications at the specified interval.
- **Notification Interval**: The period of time to wait before resending the notification.
- **Suppress Notifications**: Sets up a suppression time window during which to suspend evaluations and notifications. Useful for avoiding alarm notifications during system maintenance periods.
  - **Suppression Description**
  - **Start Time**
  - **End Time**

5. If you want to disable the new alarm, clear **Enable This Alarm?**.
6. Click **Save alarm**.
   The new alarm is listed on the **Alarm Definitions** page.

To disable or enable an alarm

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Alarm Definitions**.
2. Click the alarm that you want to disable or enable.
3. On the alarm detail page, select or clear **Alarm is Enabled**.



Note

You can also disable and enable alarms when creating or editing an alarm.

To update an alarm

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Alarm Definitions**.
2. Click the alarm that you want to update.
3. Go to **Actions** on the right, and then click **Edit Alarm**.
4. On the **Edit Alarm** page, under **Define alarm**, update alarm settings as needed:

Basic Mode (default)

By default, this page uses **Basic Mode**, which separates the metric from its dimensions and its trigger rule.

- **Alarm Name**: User-friendly name for the new alarm.
- **Alarm Severity**: The perceived type of response required when the alarm is in the firing state.
- **Alarm Body**: The human-readable content of the notification delivered. Oracle recommends providing guidance to operators for resolving the alarm condition. Consider adding links to standard runbook practices. Example: "High CPU usage alert. Follow runbook instructions for resolution."
- **Tags (optional):** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
- **Metric description**: The metric to evaluate for the alarm condition.
    - **Compartment**: The compartment containing the resources that emit the metrics evaluated by the alarm. The selected compartment is also the storage location of the alarm. By default, the first accessible compartment is selected.
    - **Metric Namespace**: The service or application emitting metrics for the resources that you want to monitor.
    - **Metric Name**: The name of the metric. Only one metric can be specified. Example: **CpuUtilization**

- **Interval**: The aggregation window, or the frequency at which data points are aggregated.

  ### Interval values

  - **1m** - 1 minute
  - **5m** - 5 minutes
  - **1h** - 1 hour

  Note

  For alarm queries, the specified interval has no effect on the resolution of the request. The only valid value of the resolution for an alarm query request is `1m`. For more information about the resolution parameter as used in alarm queries, see Alarm.

- **Statistic**: The aggregation function.

  ### Statistic values

  - **COUNT**- The number of observations received in the specified time period.
  - **MAX** - The highest value observed during the specified time period.
  - **MEAN** - The value of Sum divided by Count during the specified time period.
  - **MIN** - The lowest value observed during the specified time period.

- **P50** - The value of the 50th percentile.
- **P90** - The value of the 90th percentile.
- **P95** - The value of the 95th percentile.
- **P99** - The value of the 99th percentile.
- **P99.5** - The value of the 99.5th percentile.
- **RATE** - The per-interval average rate of change.
- **SUM** - All values added together.

- **Metric dimensions**: Optional filters to narrow the metric data evaluated.

  Dimension fields

  - **Dimension Name**: A qualifier specified in the metric definition. For example, the dimension `resourceId` is specified in the metric definition for `CpuUtilization`.

    Note

    Long lists of dimensions are trimmed.

    - To view dimensions by name, type one or more characters in the box. A refreshed (trimmed) list shows matching dimension names.
    - To retrieve all dimensions for a given metric, use the following API operation: ListMetrics

  - **Dimension Value**: The value you want to use for the specified dimension. For example, the resource identifier for your instance of interest.
  - **+ Additional dimension**: Adds another name-value pair for a dimension.

- **Trigger rule**: The condition that must be satisfied for the alarm to be in the firing state. The condition can specify a threshold, such as 90% for CPU Utilization, or an absence.

- **Operator**: The operator used in the condition threshold.

  Operator values

  - **greater than**
  - **greater than or equal to**
  - **equal to**
  - **less than**
  - **less than or equal to**
  - **between** (inclusive of specified values)
  - **outside** (inclusive of specified values)
  - **absent**

- **Value**: The value to use for the condition threshold.
- **Trigger Delay Minutes**: The number of minutes that the condition must be maintained before the alarm is in firing state.

Advanced Mode

Click **Advanced Mode** or **Switch to Advanced Mode** to view the alarm query as a Monitoring Query Language (MQL) expression. Edit your query using MQL syntax to aggregate results by group or for additional parameter values. See Monitoring Query Language (MQL) Reference.

- **Alarm Name**: User-friendly name for the new alarm.
- **Alarm Severity**: The perceived type of response required when the alarm is in the firing state.
- **Alarm Body**: The human-readable content of the notification delivered. Oracle recommends providing guidance to operators for resolving the alarm condition.

Consider adding links to standard runbook practices. Example: "High CPU usage alert. Follow runbook instructions for resolution."

- **Tags (optional):** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
- **Metric description, dimensions, and trigger rule**: The metric to evaluate for the alarm condition, including dimensions and the trigger rule.
  - **Compartment**: The compartment containing the resources that emit the metrics evaluated by the alarm. The selected compartment is also the storage location of the alarm. By default, the first accessible compartment is selected.
  - **Metric Namespace**: The service or application emitting metrics for the resources that you want to monitor.
  - **Query Code Editor** box: The alarm query as a Monitoring Query Language (MQL) expression.
    Example alarm query:

    ```
    CpuUtilization[1m]{availabilityDomain=AD1}.groupBy(poolId).percentile(0.9) > 85
    ```

    For query syntax and examples, see Working with Metric Queries.
  - **Trigger Delay Minutes**: The number of minutes that the condition must be maintained before the alarm is in firing state.

The chart below the **Define alarm** section dynamically displays the last six hours of emitted metrics according to currently selected fields for the query. Very small or large values are indicated by International System of Units (SI units), such as M for mega (10 to the sixth power).

5. Under **Notifications**, update settings as needed:

- **Destinations**:
  - **Destination Service**: The provider of the destination to use for notifications.

    Available options:
    - **Notifications Service**.
  - **Compartment**: The compartment storing the topic to be used for notifications. Can be a different compartment from the alarm and metric. By default, the first accessible compartment is selected.
  - **Topic**: The topic to use for notifications. Each topic supports a subscription protocol, such as PagerDuty.
  - **Create a topic**: Sets up a topic and subscription protocol in the selected compartment, using the specified destination service.
    - **Topic Name**: User-friendly name for the new topic. Example: "Operations Team " for a topic used to notify operations staff of firing alarms.
    - **Topic Description**: Description of the new topic.
    - **Subscription Protocol**: Medium of communication to use for the new topic. Configure your subscription for the protocol you want:

      Email subscription
      - **Subscription protocol**: Select **Email**.
      - **Email Addresses**: Type an email address.

      PagerDuty subscription
      - **Subscription Protocol**: Select **HTTPS (PagerDuty)**.

- **PagerDuty Integration URL**: Type (or copy and paste) the *integration key* portion of the URL for your PagerDuty subscription. (The other portions of the URL are hard-coded.)
  For information on setting up and retrieving your integration key, see the PagerDuty documentation.
  Example integration key: Your68a4Integration4dd8Key12f4f6
  Example integration URL: https://events.pagerduty.com/integration/Your68a4Integration4dd8Key12f4f6/enqueue

- **+ Additional destination service**: Adds another destination service and topic to use for notifications.



Note

Each alarm is limited to one destination per supported destination service.

- **Repeat Notification?**: While the alarm is in the firing state, resends notifications at the specified interval.
- **Notification Interval**: The period of time to wait before resending the notification.
- **Suppress Notifications**: Sets up a suppression time window during which to suspend evaluations and notifications. Useful for avoiding alarm notifications during system maintenance periods.
  - **Suppression Description**
  - **Start Time**
  - **End Time**

6. Select or clear **Enable this alarm?**.

7. Click **Save alarm**.
   The updated alarm settings are listed on the **Alarm Definitions** page.

To suppress alarms



Important

Only one suppression can be configured per alarm. Any existing suppression for the alarm is overwritten when you apply a new suppression.

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Alarm Definitions**.
2. On the **Alarm Definitions** page, select the check boxes for the alarms you want to suppress.



Note

You can also suppress alarms from the Alarm Status page or when creating or editing an alarm.

3. Go to **Actions** and select **Add Suppressions**.
4. In the **Suppress alarms** dialog box, select a **Start Time** and **End Time** and then optionally fill in a **Suppression Description**.
5. Click **Apply suppressions**.
   A suppression is created for each selected alarm. The updated alarm settings are listed on the **Alarm Definitions** page.

To delete alarms

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Alarm Definitions**.
2. On the **Alarm Definitions** page, select the check boxes for the alarms you want to delete.

   Note

   You can also delete an alarm from its detail page.

3. Go to **Actions** and select **Delete Alarms**.
   The deleted alarms are removed from the compartment and are no longer displayed on the **Alarm definitions** page.

To view alarm history

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Alarm Definitions**.
2. On the **Alarm Definitions** page, click the alarm that you want to view history for.
   The alarm detail page displays a chart showing data for the indicated time range and a list of timestamped transitions, such as Firing to OK.
   Alarm history is retained for 30 days.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage alarms:

- ListAlarms
- GetAlarm
- CreateAlarm
- UpdateAlarm
- DeleteAlarm
- ListAlarmsStatus
- RemoveAlarmSuppression
- GetAlarmHistory

Best Practices for your Alarms

This topic describes best practices for working with your alarms.

Create a Set of Alarms for Each Metric

For each metric emitted by your resources, create alarms that define the following resource behaviors:

- At risk. The resource is at risk of becoming inoperable, as indicated by metric values.
- Non-optimal. The resource is performing at non-optimal levels, as indicated by metric values.
- Resource is up or down. The resource is either not reachable or not operating.

The following examples use the CpuUtilization metric emitted by the oci_computeagent metric namespace. This metric monitors the utilization of the Compute instance and the activity level of any services and applications running on the instance. CpuUtilization is a key performance metric for a cloud service because it indicates CPU usage for the Compute instance and it can be used to investigate performance issues. To learn more about CPU usage, see the following URL: https://en.wikipedia.org/wiki/CPU_time.

At-Risk Example

A typical at-risk threshold for the CpuUtilization metric is any value greater than 80 percent. A Compute instance breaching this threshold is at risk of becoming inoperable. Often the cause of this behavior is one or more applications consuming a high percentage of the CPU.

In this example, you decide to notify the operations team immediately, setting the severity of the alarm as "Critical" because repair is required to bring the instances back to optimal operational levels. You configure alarm notifications to the responsible team by both PagerDuty and email, requesting an investigation and appropriate fixes before the instances go into an inoperable state. You set repeat notifications every minute. When someone responds to the alarm notifications, you temporarily stop notifications using the best practice of suppressing the alarm. Once metrics return to optimal values, you remove the suppression.

Non-Optimal Example

A typical non-optimal threshold for the CpuUtilization metric is 60 to 80 percent. When the metric values for a Compute instance are within this range, the instance is above the optimal operational range.

In this example, you decide to notify the appropriate individual or team that an application or process is consuming more CPU than usual. You configure a threshold alarm to notify the appropriate contacts, setting the severity of the alarm as "Warning," as no immediate actions are required to investigate and reduce the CPU. You set notification to email only, directed to the appropriate developer or team, with repeat notifications every 24 hours to reduce email notification noise.

Resource is Up or Down Example

A typical indicator of resource availability is a five-minute absence of the CpuUtilization metric. A Compute instance breaching this threshold is either not reachable or not operating. The resource may have stopped responding, or it might have become unavailable due to connectivity issues.

In this example, you decide to notify the operations team immediately, setting the severity of your absence alarm as "Critical" because repair is required to bring the instances online. You

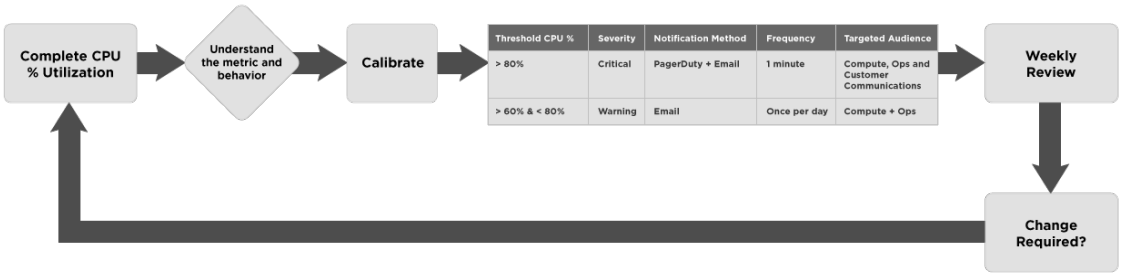
configure alarm notifications to the responsible team by both PagerDuty and email, requesting an investigation and a move of the workloads to another available resource. You set repeat notifications every minute. When someone responds to the alarm notifications, you temporarily stop notifications using the best practice of suppressing the alarm. When the CpuUtilization metric is once again detected from the resource, you remove the suppression.

Suppress Alarms During Investigations

Once a team member responds to an alarm, suppress notifications for the duration of the effort to investigate or mitigate the issue. Temporarily stopping notifications helps to avoid distractions during the investigation and mitigation. Remove the suppression when the issue has been resolved. For instructions, see To suppress alarms.

Routinely Tune Your Alarms

On a regular basis, such as weekly, review your alarms to ensure optimal configuration. Calibrate each alarm's threshold, severity, and notification details, including method, frequency, and targeted audience.



Optimal alarm configuration addresses the following factors:

- Criticality of the resource.
- Appropriate resource behavior. Assess behavior singly and within the context of the service ecosystem. Review metric value fluctuations for a given period of time and then adjust thresholds as needed.
- Acceptable notification noise. Assess the notification method (for example, email or PagerDuty), the appropriate recipients, and the frequency of repeated notifications.

For instructions, see To update an alarm.

Monitoring Query Language (MQL) Reference

This topic describes the components that appear in Monitoring Query Language (MQL) expressions, the order that they appear in, and valid values.

MQL syntax governs expressions for querying metrics that are published to the Monitoring service. In the Console, MQL expressions appear in **Advanced Mode**. If you don't need to aggregate results by group or to use other advanced query functionality, then you can create simpler versions of metric queries using **Basic Mode** in the Console.

Components in an MQL Expression

An MQL expression includes the following components:

- metric
- interval
- dimensions, as one or more name-value pairs (optional)
- grouping function (optional)
- statistic
- comparison operation (optional). Useful for defining alarms.

The query components appear in the following order (boldface components are required):

```
metric[interval]{dimensionname="dimensionvalue"}.groupingfunction.statistic
```

Comparison operation queries used for alarms can take the following formats (boldface components are required):

- **`metric[interval]`**
  `{dimensionname="dimensionvalue"}.groupingfunction.`**`statistic`** (where the statistic is `absent()`)
- **`metric[interval]`**
  `{dimensionname="dimensionvalue"}.groupingfunction.`**`statistic operator value`**
- **`metric[interval]`**
  `{dimensionname="dimensionvalue"}.groupingfunction.`**`statistic operator (value1, value2)`**

Metric Query Component

The *metric* component of the query appears before the interval.

`metric``[interval]{dimensionname="dimensionvalue"}.groupingfunction.statistic`

Valid values for *metric* depend on the resource. An example of a metric is `CpuUtilization`, sent by Compute instances. For a list of supported resources with links to their metric references, see Supported Services. You can also use the ListMetrics operation to find metrics sent by a particular service, such as the Compute service. This operation returns metric definitions.

Interval Query Component

The *interval* component of the query appears between the metric and statistic (before the optional dimension name-value pair and grouping function).

`metric[`*`interval`*`]{dimensionname="dimensionvalue"}.groupingfunction.statistic`

The Monitoring Query Language (MQL) syntax (**Advanced Mode** in the Console) supports the following range of values for *interval*:

`1m-60m` (also, `1h`)

The **Interval** option in the Console (**Basic Mode**) supports the following range of values:

- **1m** - 1 minute
- **5m** - 5 minutes
- **1h** - 1 hour



Note

For metric queries, the interval you select drives the default resolution of the request, which determines the maximum time range of data returned.

Maximum time range returned for a query

The maximum time range returned for a metric query depends on the resolution. By default, for metric queries, the resolution is the same as the query interval. Following are the maximum time ranges returned for each interval selection available in the Console.

Interval	Default resolution (metric queries)	Maximum time range returned
1h	1 hour	90 days
5m	5 minutes	30 days
1m	1 minute	7 days

For more information about the resolution parameter as used in metric queries, see SummarizeMetricsData.

For alarm queries, the specified interval has no effect on the resolution of the request. The only valid value of the resolution for an alarm query request is `1m`. For more information about the resolution parameter as used in alarm queries, see Alarm.

Dimension Query Component

The *dimensionname="dimensionvalue"* component of the query appears between the interval and statistic (before the optional grouping function).

`metric[interval]{dimensionname="dimensionvalue"}.groupingfunction.statistic`

Surround the dimension value with double quotes. Example dimension name-value pair for filtering by availability domain: `availabilityDomain = "VeBZ:PHX-AD-1"`

You can specify multiple dimension name-value pairs. Place each pair within the brackets and separate the pairs with commas.

Valid values for *dimensionname* depend on the *metric*. An example of a dimension name is `resourceDisplayName`, included with the `CpuUtilization` metric sent by Compute instances. For a list of supported resources with links to their metric references, including dimensions, see Supported Services. You can also use the ListMetrics operation to find metrics (and their dimensions) sent by a particular application or service, such as the Compute service.

Grouping Function Query Component

The *groupingfunction* component of the query appears between the interval and statistic (after the optional dimension name-value pair).

`metric[interval]{dimensionname="dimensionvalue"}.groupingfunction.statistic`

Valid grouping functions are as follows.

Grouping function (MQL expression; Advanced Mode in the Console)	Grouping function option (Basic Mode in the Console)	Description
`groupBy()`	(not available)	Aggregates query results by group (dimension). For example, `groupBy (availabilityDomain)` groups results by availability domain so that results from each availability domain are together.
`grouping()`	**Aggregate Metric Streams**	Aggregates all query results.

Statistic Query Component

The *statistic* component of the query appears after the interval and optional dimension name-value pair and grouping function.

`metric[interval]{dimensionname="dimensionvalue"}.groupingfunction.`*`statistic`*

Valid statistics are as follows.

Statistic (MQL expression; Advanced Mode in the Console)	Statistic option (Basic Mode in the Console)	Description
`absent()`	(see absent)	Sets the trigger condition as absence of the specified metric. (Applies to comparison operation queries only. Useful for defining alarms.)
`avg()`	(not available)	Returns the value of Sum divided by Count during the specified time period. Identical to `mean()`.
`count()`	**COUNT**	Returns the number of observations received in the specified time period.
`increment()`	(not available)	Returns the per-interval change.
`max()`	**MAX**	Returns the highest value observed during the specified time period.
`mean()`	**MEAN**	Returns the value of Sum divided by Count during the specified time period.
`min()`	**MIN**	Returns the lowest value observed during the specified time period.

Statistic (MQL expression; Advanced Mode in the Console)	Statistic option (Basic Mode in the Console)	Description
`percentile()`	**P50** **P90** **P95** **P99** **P99.9**	Returns the estimated value of the specified percentile. Valid values are greater than 0.0 and less than 1.0. For example, `percentile(0.8)` returns the value of the 80th percentile.
`rate()`	**RATE**	Returns the per-interval average rate of change. The unit is per-second.
`sum()`	**SUM**	Returns all values added together.

Operator and Value Query Component

The *operator value* component of the query appears after the statistic in threshold alarm queries. Either one or two values are needed, depending on the operator:

- `metric[interval]{dimensionname="dimensionvalue"}.groupingfunction.statistic` *`operator value`*
- `metric[interval]{dimensionname="dimensionvalue"}.groupingfunction.statistic` *`operator (value1, value2)`*

Valid operators are as follows.

Operator (MQL expression; Advanced Mode in the Console)	Operator option (Basic Mode in the Console)	Number of values
`>`	**greater than**	1
`>=`	**greater than or equal to**	1
`==`	**equal to**	1
`!=` (not equal to)	(not available)	1
`<`	**less than**	1
`<=`	**less than or equal to**	1
`in` (inclusive of specified values)	**between** (inclusive of specified values)	2
`not in` (inclusive of specified values)	**outside** (inclusive of specified values)	2
Not applicable. See absent().	**absent**	0

CHAPTER 18 Networking

This chapter explains how to set up cloud networks.

Overview of Networking

When you work with Oracle Cloud Infrastructure, one of the first steps is to set up a virtual cloud network (VCN) for your cloud resources. This topic gives you an overview of Oracle Cloud Infrastructure Networking components and typical scenarios for using a VCN.

Networking Components

The Networking service uses virtual versions of traditional network components you might already be familiar with:

VIRTUAL CLOUD NETWORK (VCN)

A virtual, private network that you set up in Oracle data centers. It closely resembles a traditional network, with firewall rules and specific types of communication gateways that you can choose to use. A VCN resides in a single Oracle Cloud Infrastructure region and covers a single, contiguous IPv4 CIDR block of your choice. See Allowed VCN Size and Address Ranges. The terms *virtual cloud network*, *VCN, and cloud network* are used interchangeably in this documentation. For more information, see VCNs and Subnets.

SUBNETS

Subdivisions you define in a VCN (for example, 10.0.0.0/24 and 10.0.1.0/24). Subnets contain virtual network interface cards (VNICs), which attach to instances. Each subnet consists of a contiguous range of IP addresses that do not overlap with other subnets in the VCN. You can designate a subnet to exist either in a single availability domain or across an entire region (regional subnets are recommended). Subnets act as a unit of configuration within the VCN: All VNICs in a given subnet use the same route table, security lists, and DHCP options (see the definitions that follow). You can designate a subnet as either public or private when you create it. Private means VNICs in the subnet

can't have public IP addresses. Public means VNICs in the subnet can have public IP addresses at your discretion. See Access to the Internet.

VNIC

A virtual network interface card (VNIC), which attaches to an instance and resides in a subnet to enable a connection to the subnet's VCN. The VNIC determines how the instance connects with endpoints inside and outside the VCN. Each instance has a primary VNIC that's created during instance launch and cannot be removed. You can add secondary VNICs to an existing instance (in the same availability domain as the primary VNIC), and remove them as you like. Each secondary VNIC can be in a subnet in the same VCN as the primary VNIC, or in a different subnet that is either in the same VCN or a different one. However, all the VNICs must be in the same availability domain as the instance. For more information, see Virtual Network Interface Cards (VNICs).

PRIVATE IP

A private IP address and related information for addressing an instance (for example, a hostname for DNS). Each VNIC has a primary private IP, and you can add and remove secondary private IPs. The primary private IP address on an instance doesn't change during the instance's lifetime and cannot be removed from the instance. For more information, see Private IP Addresses.

PUBLIC IP

A public IP address and related information. You can optionally assign a public IP to your instances or other resources that have a private IP. Public IPs can be either *ephemeral* or *reserved*. For more information, see Public IP Addresses.

DYNAMIC ROUTING GATEWAY (DRG)

An optional virtual router that you can add to your VCN. It provides a path for *private* network traffic between your VCN and on-premises network. You can use it with other Networking components and a router in your on-premises network to establish a connection by way of IPSec VPN or Oracle Cloud Infrastructure FastConnect. It can also provide a path for private network traffic between your VCN and another VCN in a

different region. For more information, see Access to Your On-Premises Network, Dynamic Routing Gateways (DRGs), and Remote VCN Peering (Across Regions).

INTERNET GATEWAY

Another optional virtual router that you can add to your VCN for direct internet access. For more information, see Access to the Internet and also Scenario A: Public Subnet.

NETWORK ADDRESS TRANSLATION (NAT) GATEWAY

Another optional virtual router that you can add to your VCN. It gives cloud resources without public IP addresses access to the internet without exposing those resources to incoming internet connections. For more information, see Access to the Internet and also NAT Gateway.

SERVICE GATEWAY

Another optional virtual router that you can add to your VCN. It provides a path for *private* network traffic between your VCN and supported services in the Oracle Services Network (examples: Oracle Cloud Infrastructure Object Storage and Autonomous Database). For example, DB Systems in a private subnet in your VCN can back up data to Object Storage without needing public IP addresses or access to the internet. For more information, see Access to Oracle Services: Service Gateway.

LOCAL PEERING GATEWAY (LPG)

Another optional virtual router that you can add to your VCN. It lets you peer one VCN with another VCN in the same region. *Peering* means the VCNs communicate using private IP addresses, without the traffic traversing the internet or routing through your on-premises network. A given VCN must have a separate LPG for each peering it establishes. For more information, see Local VCN Peering (Within Region).

REMOTE PEERING CONNECTION (RPC)

A component that you can add to a DRG. It lets you peer one VCN with another VCN in a *different* region. For more information, see Remote VCN Peering (Across Regions).

ROUTE TABLES

Virtual route tables for your VCN. They have rules to route traffic from subnets to destinations outside the VCN by way of gateways or specially configured instances. Your VCN comes with an empty default route table, and you can add custom route tables of your own. For more information, see Route Tables.

SECURITY LISTS

Virtual firewall rules for your VCN. Security lists have ingress and egress rules that specify the types of traffic (protocol and port) allowed in and out of the instances. You can choose whether a given rule is stateful or stateless. For example, you can allow incoming SSH traffic from anywhere to a subnet's instances by setting up a stateful ingress rule with source CIDR 0.0.0.0/0, and destination TCP port 22. Your VCN comes with a default security list with default rules, and you can add custom security lists of your own. For more information, see Security Lists.

DHCP OPTIONS

Configuration information that is automatically provided to the instances when they boot up. For more information, see DHCP Options.

Allowed VCN Size and Address Ranges

A VCN covers a single, contiguous IPv4 CIDR block of your choice. The allowable VCN size range is /16 to /30. Example: 10.0.0.0/16. The Networking service reserves the first two IP addresses and the last one in each subnet's CIDR. After you've created a VCN or subnet, you can't change its size, so it's important to think about the size of VCN and subnets you need before creating them.

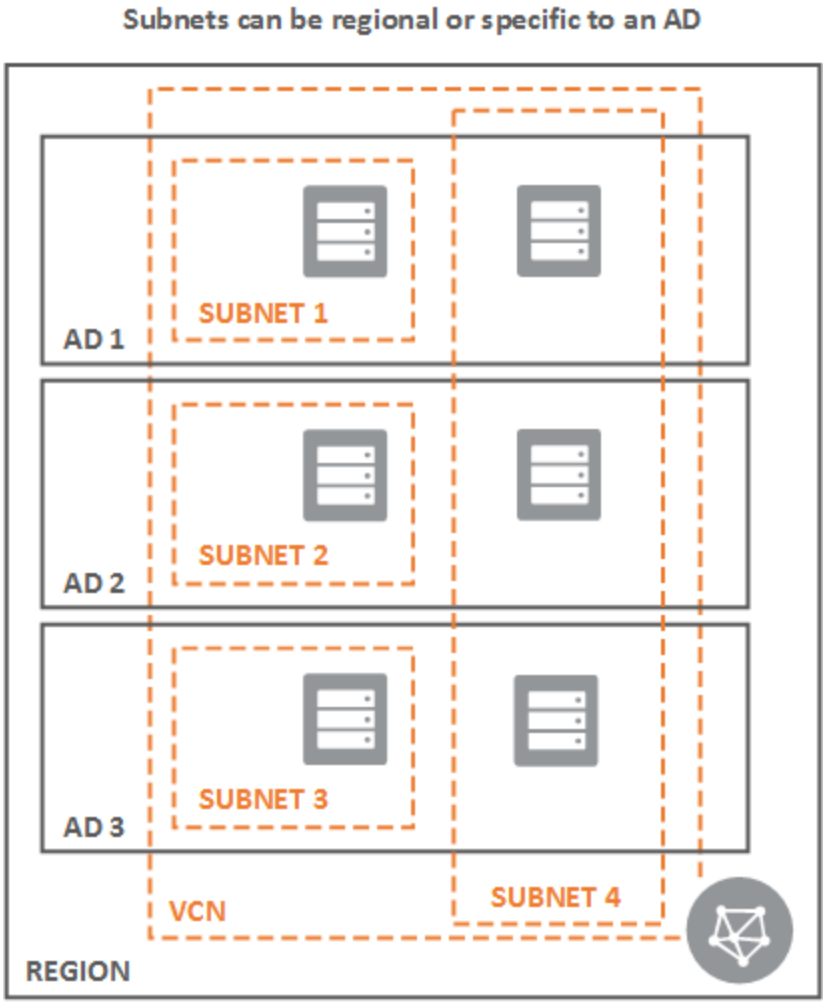
For your VCN, Oracle recommends using one of the private IP address ranges specified in RFC 1918 (10.0.0.0/8, 172.16/12, and 192.168/16). However, you can use a publicly routable range. Regardless, this documentation uses the term *private IP address* when referring to IP addresses in your VCN's CIDR.

The VCN's CIDR must not overlap with your on-premises network or another VCN you peer with. The subnets in a given VCN must not overlap with each other. For reference, here's a CIDR calculator.

Availability Domains and Your VCN

Your VCN resides in a single Oracle Cloud Infrastructure region. A region can have multiple availability domains to provide isolation and redundancy. For more information, see Regions and Availability Domains.

Originally subnets were designed to cover only one availability domain (AD) in a region. They were all *AD-specific*, which means the subnet's resources were required to reside in a particular availability domain. Now subnets can be either AD-specific or *regional*. You choose the type when you create the subnet. Both types of subnets can co-exist in the same VCN. In the following diagram, subnets 1-3 are AD-specific, and subnet 4 is regional.



Aside from the removal of the AD constraint, regional subnets behave the same as AD-specific subnets. **Oracle recommends using regional subnets** because they're more flexible. They make it easier to efficiently divide your VCN into subnets while also designing for availability domain failure.

When you create a resource such as a Compute instance, you choose which availability domain the resource will be in. From a virtual networking standpoint, you must also choose

which VCN and subnet the instance will be in. You can either choose a regional subnet, or choose an AD-specific subnet that matches the AD you chose for the instance.

Default Components that Come With Your VCN

Your VCN automatically comes with these *default* components:

- Default route table, with no rules
- Default security list, with default rules
- Default set of DHCP options, with default values

You can't delete these default components. However, you can change their contents (for example, the rules in the default security list). And you can create your own custom versions of each kind of component in your VCN. There are limits to how many you can create and the maximum number of rules. For more information, see Service Limits.

Each subnet always has these components associated with it:

- One route table
- One or more security lists (for the maximum number, see Service Limits)
- One set of DHCP options

During subnet creation, you can choose which route table, security list, and set of DHCP options the subnet uses. If you don't specify a particular component, the subnet automatically uses the VCN's default component. You can change which components the subnet uses at any time.

Connectivity Choices

You can control whether subnets are public or private, and whether instances get public IP addresses. You can set up your VCN to have access to the internet if you like. You can also privately connect your VCN to public Oracle Cloud Infrastructure services such as Object Storage, to your on-premises network, or to another VCN.

Public vs. Private Subnets

When you create a subnet, by default it's considered public, which means instances in that subnet are allowed to have public IP addresses. Whoever launches the instance chooses whether it will have a public IP address. You can override that behavior when creating the subnet and request that it be private, which means instances launched in the subnet are prohibited from having public IP addresses. Network administrators can therefore ensure that instances in the subnet have no internet access, even if the VCN has a working internet gateway, and security lists and firewall rules allow the traffic.

How IP Addresses Are Assigned

Each instance has a primary VNIC that's created during instance launch and cannot be removed. You can add secondary VNICs to an existing instance (in the same availability domain as the primary VNIC) and remove them as you like.

Every VNIC has a private IP address from the associated subnet's CIDR. You can choose the particular IP address (during instance launch or secondary VNIC creation), or Oracle can choose it for you. The private IP address does not change during the lifetime of the instance and cannot be removed. You can also add secondary private IPs to a VNIC.

If the VNIC is in a public subnet, then each private IP on that VNIC can have a public IP assigned to it at your discretion. Oracle chooses the particular IP address. There are two types of public IPs: *ephemeral* and *reserved*. An ephemeral public IP exists only for the lifetime of the private IP it's assigned to. In contrast, a reserved public IP exists as long as you want it to. You maintain a pool of reserved public IPs and allocate them to your instances at your discretion. You can move them from resource to resource in a region as you need to.

Access to the Internet

There are two optional gateways (virtual routers) that you can add to your VCN depending on the type of internet access you need:

- **Internet gateway:** For resources with public IP addresses that need to be reached from the internet (example: a web server) or need to initiate connections to the internet.

- **NAT gateway:** For resources without public IP addresses that need to initiate connections to the internet (example: for software updates) but need to be protected from inbound connections from the internet.

Just having an internet gateway alone does not expose the instances in the VCN's subnets directly to the internet. The following requirements must also be met:

- The internet gateway must be enabled (by default, the internet gateway is enabled upon creation).
- The subnet must be public.
- The subnet must have a route rule that directs traffic to the internet gateway.
- The subnet must have security list rules that allow the traffic (and each instance's firewall must allow the traffic).
- The instance must have a public IP address.



Tip

To access public services such as Object Storage from your VCN without the traffic going over the internet, use a service gateway.

Also, be aware that when an internet gateway receives traffic from your VCN destined for a public IP address *that is part of Oracle Cloud Infrastructure* (such as Object Storage), the internet gateway routes the traffic to the destination without sending the traffic over the internet.

You can also give a subnet *indirect* access to the internet by setting up an internet proxy in your on-premises network and then connecting that network to your VCN by way of a DRG. For more information, see Access to Your On-Premises Network.

Access to Public Oracle Cloud Infrastructure Services

You can use a service gateway with your VCN to enable private access to public Oracle Cloud Infrastructure services such as Object Storage. For example, DB Systems in a private subnet in your VCN can back up data to Object Storage without needing public IP addresses or access to the internet. No internet gateway or NAT is required. For more information, see Access to Oracle Services: Service Gateway.

Access to Your On-Premises Network

There are two ways to connect your on-premises network to Oracle Cloud Infrastructure:

- VPN Connect: Offers multiple IPSec tunnels between your existing network's edge and your VCN, by way of a DRG that you create and attach to your VCN.
- Oracle Cloud Infrastructure FastConnect: Offers a private connection between your existing network's edge and Oracle Cloud Infrastructure. Traffic does not traverse the internet. Both private peering and public peering are supported. That means your on-premises hosts can access private IPv4 addresses in your VCN as well as regional public IPv4 addresses in Oracle Cloud Infrastructure (for example, Object Storage or public load balancers in your VCN).

You can use one or both types of the preceding connections. If you use both, you can use them simultaneously, or in a redundant configuration. These connections come to your VCN by way of a single DRG that you create and attach to your VCN. Without that DRG attachment and a route rule for the DRG, traffic does not flow between your VCN and on-premises network. At any time, you can detach the DRG from your VCN but maintain all the remaining components that form the rest of the connection. You could then reattach the DRG again, or attach it to another VCN.

Access to Another VCN

You can connect your VCN to another VCN over a private connection that doesn't require the traffic to traverse the internet. In general, this type of connection is referred to as *VCN peering*. Each VCN must have specific components to enable peering. The VCNs must also have specific IAM policies, route rules, and security lists that permit the connection to be made and the desired network traffic to flow over the connection. For more information, see Access to Other VCNs: Peering.

Connection to Oracle Cloud Infrastructure Classic

You can set up a connection between your Oracle Cloud Infrastructure environment and Oracle Cloud Infrastructure Classic environment. This connection can facilitate hybrid deployments between the two environments, or migration from Oracle Cloud Infrastructure Classic to Oracle Cloud Infrastructure. For more information, see Access to Oracle Cloud Infrastructure Classic.

Connection to Other Clouds with Libreswan

You can connect your VCN to another cloud provider by using an IPSec VPN with a Libreswan VM as the customer-premises equipment (CPE). For more information, see Access to Other Clouds with Libreswan.

Networking Scenarios

This documentation includes a few basic networking scenarios to help you understand the Networking service and generally how the components work together. See these topics:

- Scenario A: Public Subnet
- Scenario B: Private Subnet with a VPN
- Scenario C: Public and Private Subnets with a VPN

There's also an advanced scenario that enables communication between an on-premises network and multiple VCNs over a single Oracle Cloud Infrastructure FastConnect or IPSec VPN. For more information, see Advanced Scenario: Transit Routing.

Regions and Availability Domains

Your VCN resides in a single Oracle Cloud Infrastructure region. Each subnet resides in a single availability domain (AD). Availability domains are designed to provide isolation and redundancy in your VCN, as illustrated in Scenario B and C earlier. For example, you could set up your primary set of subnets in a single AD, and then set up a duplicate set of subnets in a secondary AD. The two ADs are isolated from each other in the Oracle data centers, so if one fails, you can easily switch over to the other AD. For more information, see Regions and Availability Domains.

Public IP Address Ranges

For a list of Oracle Cloud Infrastructure public IP ranges to whitelist, see IP Address Ranges.

IP Addresses Reserved for Use by Oracle

Certain IP addresses are reserved for Oracle Cloud Infrastructure use and may not be used in your address numbering scheme.

169.254.0.0/16

These addresses are used for iSCSI connections to the boot and block volumes, instance metadata, and other services.

Three IP Addresses in Each Subnet

These addresses consist of:

- The first IP address in the CIDR (the network address)
- The last IP address in the CIDR (the broadcast address)
- The first host address in the CIDR (the subnet default gateway address)

For example, in a subnet with CIDR 192.168.0.0/24, these addresses are reserved:

- 192.168.0.0 (the network address)
- 192.168.0.255 (the broadcast address)
- 192.168.0.1 (the subnet default gateway address)

The remaining addresses in the CIDR (192.168.0.2 to 192.168.0.254) are available for use.

Resource Identifiers

Most types of Oracle Cloud Infrastructure resources have a unique, Oracle-assigned identifier called an Oracle Cloud ID (OCID). For information about the OCID format and other ways to identify your resources, see Resource Identifiers.

Ways to Access Oracle Cloud Infrastructure

You can access Oracle Cloud Infrastructure using the Console (a browser-based interface) or the REST API. Instructions for the Console and API are included in topics throughout this guide. For a list of available SDKs, see Software Development Kits and Command Line Interface.

To access the Console, you must use a supported browser. Oracle Cloud Infrastructure supports the latest desktop versions of Google Chrome, Microsoft Edge, Internet Explorer 11, Safari, Firefox, and Firefox ESR. Note that private browsing mode is not supported for Firefox, Internet Explorer, or Edge. Mobile browsers are not supported.

For general information about using the API, see REST APIs.

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see

Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

Limits on Your Networking Components

See Service Limits for a list of applicable limits and instructions for requesting a limit increase.

Scenario A: Public Subnet

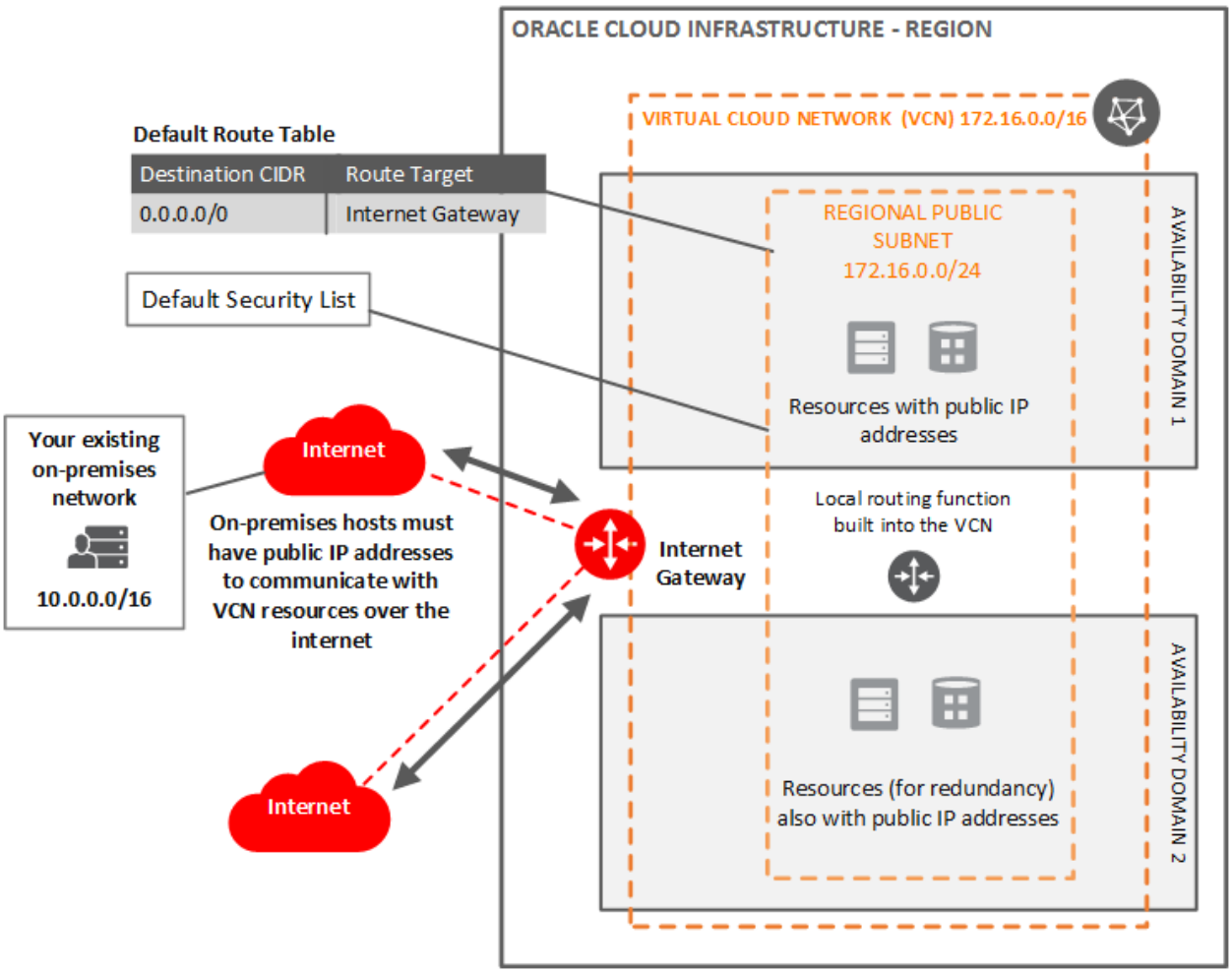
This topic explains how to set up Scenario A, which consists of a virtual cloud network (VCN) and a regional public subnet. There are public servers in separate availability domains for redundancy. The VCN is directly connected to the internet by way of an internet gateway. The gateway is also used for connectivity to your on-premises network. Any resource in the on-premises network that needs to communicate with resources in the VCN must have a public IP address and access to the internet.

The subnet uses the default security list, which has default rules that are designed to make it easy to get started with Oracle Cloud Infrastructure. The rules enable typical required access (for example, inbound SSH connections and any type of outbound connections). Remember that security list rules only *allow* traffic. Any traffic not explicitly covered by a security list rule is implicitly denied.

In this scenario, you add additional rules to the default security list. You could instead create a custom security list for those rules. You would then set up the subnet to use both the default security list and the custom security list.

The subnet uses the default route table, which starts out with no rules when the VCN is created. In this scenario, the table has only a single rule for the internet gateway.

See the following figure.



Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

If you're a member of the Administrators group, you already have the required access to execute Scenario A. Otherwise, you need access to Networking, and you need the ability to launch instances. See IAM Policies for Networking.

Setting Up Scenario A in the Console

Setup is easy in the Console.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Task 1: Create the VCN

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Choose a compartment you have permission to work in (on the left side of the page). The page updates to display only the resources in that compartment. If you're not sure which compartment to use, contact an administrator. For more information, see Access Control.
3. Click **Create Virtual Cloud Network**.
4. Enter the following:
    a. **Name:** A friendly name for the VCN. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
    b. **Create in Compartment:** Leave as is.

c. **Create Virtual Cloud Network Only:** Make sure this radio button is selected (the default).

d. **CIDR Block:** A single, contiguous CIDR block for the VCN. For example: 172.16.0.0/16. You *cannot* change this value later. See Allowed VCN Size and Address Ranges. For reference, here's a CIDR calculator.

e. **Use DNS Hostnames in this VCN:** If you want the instances in the VCN to have DNS hostnames (which can be used with the built-in DNS capability in the VCN), select the **Use DNS Hostnames in this VCN** check box. Then you can specify a DNS label for the VCN, or the Console will generate one for you. The dialog box automatically displays the corresponding **DNS Domain Name** for the VCN (*<VCN DNS label>*.oraclevcn.com). For more information, see DNS in Your Virtual Cloud Network.

f. **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

5. Click **Create Virtual Cloud Network**.
   The VCN is then created and displayed on the **Virtual Cloud Networks** page in the compartment you chose.

Task 2: Create the regional public subnet

1. While still viewing the VCN, click **Create Subnet**.
2. Enter the following:
   - **Name:** A friendly name for the subnet (for example, Regional Public Subnet). It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.

- **Regional or Availability Domain-Specific:** Select **Regional** (recommended), which means the subnet spans all availability domains in the region. Later when you launch an instance, you can create it in any availability domain in the region. For more information, see About Regional Subnets.
- **CIDR Block:** A single, contiguous CIDR block within the VCN's CIDR block. For example: 172.16.0.0/24. You *cannot* change this value later. For reference, here's a CIDR calculator.
- **Route Table:** Select the default route table.
- **Private or public subnet**: Select **Public Subnet**, which means instances in the subnet can optionally have public IP addresses. For more information, see Access to the Internet.
- **Use DNS Hostnames in this Subnet:**This option is available only if you provided a DNS label for the VCN during creation. If you want this subnet's instances to have DNS hostnames (which can be used with the built-in DNS capability in the VCN), select the check box for **Use DNS Hostnames in this Subnet**. Then you may specify a DNS label for the subnet, or the Console will generate one for you. The dialog box will automatically display the corresponding **DNS Domain Name** for the subnet (*<subnet DNS label>*.*<VCN DNS label>*.oraclevcn.com). For more information, see DNS in Your Virtual Cloud Network.
- **DHCP Options:** Select the default set of DHCP options.
- **Security Lists:** Make sure the default security list is selected (the default).
- **Tags:** Leave as is. You can add tags later if you want. For more information, see Resource Tags.

3. Click **Create Subnet**.
   The subnet is then created and displayed on the **Subnets** page.

Task 3: Create the internet gateway

1. Under **Resources**, click **Internet Gateways**.

2. Click **Create Internet Gateway**.
3. Enter the following:
    - **Name:** A friendly name for the internet gateway. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
    - **Create in Compartment:** Leave as is.
    - **Tags:** Leave as is. You can add tags later if you want. For more information, see Resource Tags.
4. Click **Create Internet Gateway**.
   Your internet gateway is created and displayed on the **Internet Gateways** page. It's already enabled, but you must add a route rule that allows traffic to flow to the gateway.

Task 4: Update the default route table to use the internet gateway

The default route table starts out with no rules. Here you add a rule that routes all traffic destined for addresses outside the VCN to the internet gateway. The existence of this rule also enables inbound connections to come from the internet to the subnet, through the internet gateway. You use security list rules to control the *types of traffic* that are allowed in and out of the instances in the subnet (see the next task).

No route rule is required in order to route traffic within the VCN itself.

1. Under **Resources**, click **Route Tables**.
2. Click the default route table to view its details.
3. Click **Add Route Rule**.
4. Enter the following:
    - **Target Type:** Internet Gateway

- **Destination CIDR block:** 0.0.0.0/0 (which means that all non-intra-VCN traffic that is not already covered by other rules in the route table goes to the target specified in this rule)
- **Compartment:** The compartment where the internet gateway is located.
- **Target:** The internet gateway you created.

5. Click **Add Route Rule**.

The default route table now has a rule for the internet gateway. Because the subnet was set up to use the default route table, the resources in the subnet can now use the internet gateway. The next step is to specify the types of traffic you want to allow in and out of the instances you later create in the subnet.

Task 5: Update the default security list

Earlier you set up the subnet to use the VCN's default security list. Now you add security list rules that allow the types of connections that the instances in the VCN will need.

For example: This is a public subnet with an internet gateway, so the instances your launch might need to receive inbound HTTPS connections from the internet (if they're web servers). Here's how to add another rule to the default security list to enable that traffic:

1. Under **Resources**, click **Security Lists.**
2. Click the default security list to view its details. By default, you land on the **Ingress Rules** page.
3. Click **Add Ingress Rule**.
4. To enable inbound connections for HTTPS (TCP port 443), enter the following:
   - **Stateless**: Unselected (this is a stateful rule)
   - **Source Type:** CIDR
   - **Source CIDR:** 0.0.0.0/0
   - **IP Protocol:** TCP

- **Source Port Range:** All
- **Destination Port Range**: 443

5. Click **Add Ingress Rule**.

> **Important**
>
> *Security List Rule for Windows Instances*
>
> If you're going to launch Windows instances, you need to add a security list rule to enable Remote Desktop Protocol (RDP) access. Specifically, you need a stateful ingress rule for TCP traffic on destination port 3389 from source 0.0.0.0/0 and any source port. For more information, see Security Lists.

For a production VCN, you typically set up one or more *custom* security lists for each subnet. If you like, you can edit the subnet to use different security lists. If you choose not to use the default security list, do so only after carefully assessing which of its default rules you want to duplicate in your custom security list. For example: the default ICMP rules in the default security list are important for receiving connectivity messages.

Task 6: Create instances in separate availability domains

Your next step is to create one or more instances in the subnet. The scenario's diagram shows instances in two different availability domains. When you create the instance, you choose the AD, which VCN and subnet to use, and several other characteristics.

Each instance automatically gets a private IP address. When you create an instance in a *public subnet*, you choose whether the instance gets a public IP address. With this network setup in Scenario A, you *must* give each instance a public IP address, or else you can't access them through the internet gateway. The default (for a public subnet) is for the instance to get a public IP address.

After creating an instance in this scenario, you can connect to it over the internet with SSH or RDP from your on-premises network or other location on the internet. For more information and instructions, see Launching an Instance.

Setting Up Scenario A with the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operations:

1. CreateVcn: Make sure to include a DNS label for the VCN if you want the instances to have hostnames (see DNS in Your Virtual Cloud Network).
2. CreateSubnet: Create one regional public subnet. Include a DNS label for the subnet if you want the instances to have hostnames. Use the default route table, default security list, and default set of DHCP options.
3. CreateInternetGateway
4. UpdateRouteTable: To enable communication with the internet gateway, update the default route table to include a route rule with destination = 0.0.0.0/0, and destination target = the internet gateway. This rule routes all traffic destined for addresses outside the VCN to the internet gateway. No route rule is required in order to route traffic within the VCN itself.
5. UpdateSecurityList: To allow specific types of connections to and from the instances in the subnet.



Important

Security List Rule for Windows Instances

If you're going to launch Windows instances, you need to add a security list rule to enable Remote Desktop Protocol (RDP) access. Specifically, you need a stateful ingress rule for TCP traffic on destination port 3389 from source 0.0.0.0/0 and any source port. For more information, see Security Lists.

Your next step is to create one or more instances in the subnet. The scenario's diagram shows instances in two different availability domains. When you create the instance, you choose the AD, which VCN and subnet to use, and several other characteristics.

Each instance automatically gets a private IP address. When you create an instance in a *public subnet*, you choose whether the instance gets a public IP address. With this network setup in Scenario A, you *must* give each instance a public IP address, or else you can't access them through the internet gateway. The default (for a public subnet) is for the instance to get a public IP address.

After creating an instance in this scenario, you can connect to it over the internet with SSH or RDP from your on-premises network or other location on the internet. For more information and instructions, see Launching an Instance.

Scenario B: Private Subnet with a VPN

This topic explains how to set up Scenario B, which consists of a virtual cloud network (VCN) with a regional private subnet. There are servers in separate availability domains for redundancy. The VCN has a dynamic routing gateway (DRG) and IPSec VPN for connectivity to your on-premises network. The VCN has no direct connection to the internet. Any connection to the internet would need to come indirectly by way of the on-premises network.

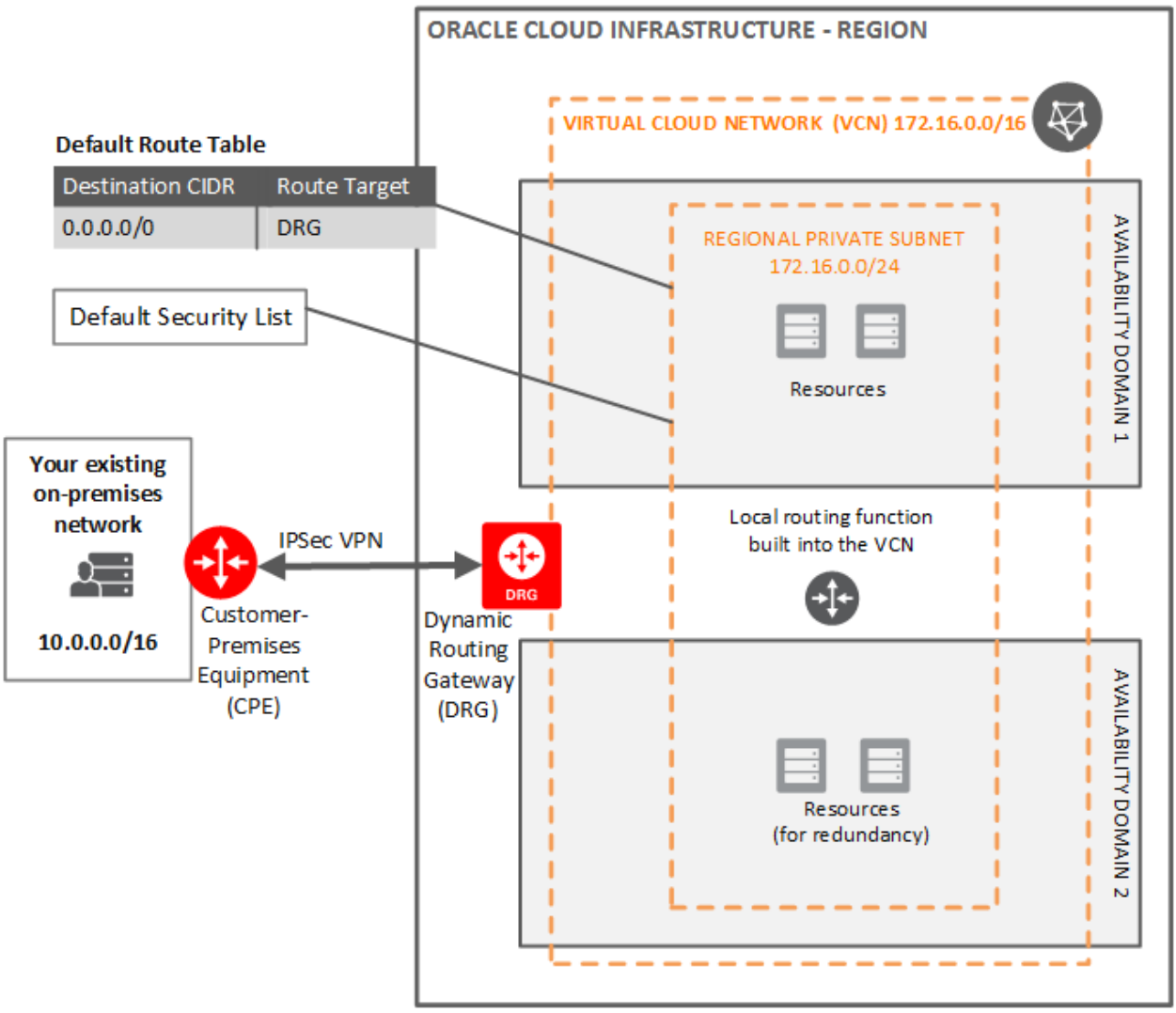
The subnet uses the default security list, which has default rules that are designed to make it easy to get started with Oracle Cloud Infrastructure. The rules enable typical required access

(for example, inbound SSH connections and any type of outbound connections). Remember that security list rules only *allow* traffic. Any traffic not explicitly covered by a security list rule is denied.

In this scenario, you add additional rules to the default security list. You could instead create a custom security list for those rules. You would then set up the subnet to use both the default security list and the custom security list.

The subnet uses the default route table, which starts out with no rules when the VCN is created. In this scenario, the table has only a single rule for the DRG. No route rule is required in order to route traffic within the VCN itself.

See the following figure.


ORACLE CLOUD INFRASTRUCTURE - REGION
VIRTUAL CLOUD NETWORK (VCN) 172.16.0.0/16
Default Route Table
Destination CIDR
Route Target
0.0.0.0/0
DRG
Default Security List
REGIONAL PRIVATE SUBNET
172.16.0.0/24
Resources
AVAILABILITY DOMAIN 1
Your existing on-premises network
10.0.0.0/16
IPSec VPN
Customer-Premises Equipment (CPE)
DRG
Dynamic Routing Gateway (DRG)
Local routing function built into the VCN
Resources
(for redundancy)
AVAILABILITY DOMAIN 2



Tip

The scenario uses an IPSec VPN for connectivity. However, you could instead use Oracle Cloud InfrastructureFastConnect.

Prerequisites

To set up the VPN in this scenario, you need to get the following information from a network administrator:

- Public IP address of the customer-premises equipment (CPE) at your end of the VPN
- Static routes for your on-premises network

You will provide Oracle this information and in return receive the information your network administrator must have to configure the CPE at your end of the VPN.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

If you're a member of the Administrators group, you already have the required access to execute Scenario B. Otherwise, you need access to Networking, and you need the ability to launch instances. See IAM Policies for Networking.

Setting Up Scenario B

Setup is easy in the Console. Alternatively, you can use the Oracle Cloud Infrastructure API, which lets you execute the individual operations yourself.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.



Important

Most of this process involves working with the Console or API (whichever you choose) for a short period to set up the desired Networking components. But there's also a critical step that requires a network administrator in your organization to take information you receive from setting up the components and use it to configure the CPE at your end of the VPN. Therefore you can't complete this process in one short session. You'll need to break for an unknown period of time while the network administrator completes the configuration and then return afterward to confirm communication with your instances over the VPN.

Using the Console

Task 1: Set up the VCN and subnet

1. Create the VCN:
    a. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.

b. Choose a compartment you have permission to work in (on the left side of the page). The page updates to display only the resources in that compartment. If you're not sure which compartment to use, contact an administrator. For more information, see Access Control.

c. Click **Create Virtual Cloud Network**.

d. Enter the following:

- **Name:** A friendly name for the VCN. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
- **Create in Compartment:** Leave as is.
- **Create Virtual Cloud Network Only:** Make sure this radio button is selected (the default).
- **CIDR Block:** A single, contiguous CIDR block for the VCN. For example: 172.16.0.0/16. You *cannot* change this value later. See Allowed VCN Size and Address Ranges. For reference, here's a CIDR calculator.
- **Use DNS Hostnames in this VCN:** If you want the instances in the VCN to have DNS hostnames (which can be used with the built-in DNS capability in the VCN), select the **Use DNS Hostnames in this VCN** check box. Then you can specify a DNS label for the VCN, or the Console will generate one for you. The dialog box automatically displays the corresponding **DNS Domain Name** for the VCN (`<VCN DNS label>.oraclevcn.com`). For more information, see DNS in Your Virtual Cloud Network.
- **Tags:** Leave as is. You can add tags later if you want. For more information, see Resource Tags.

e. Click **Create Virtual Cloud Network**.
The VCN is then created and displayed on the **Virtual Cloud Networks** page in the compartment you chose.

2. Create the regional private subnet:

a. While still viewing the VCN, click **Create Subnet**.

b. Enter the following:

- **Name:** A friendly name for the subnet (for example, *Regional Private Subnet*). It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
- **Regional or Availability Domain-Specific:** Select **Regional** (recommended), which means the subnet spans all availability domains in the region. Later when you launch an instance, you can create it any availability domain in the region. For more information, see About Regional Subnets.
- **CIDR Block:** A single, contiguous CIDR block within the VCN's CIDR block. For example: 172.16.0.0/24. You *cannot* change this value later. For reference, here's a CIDR calculator.
- **Route Table:** Select the default route table.
- **Private or public subnet**: Select **Private Subnet**, which means instances in the subnet cannot have public IP addresses. For more information, see Access to the Internet.
- **Use DNS Hostnames in this Subnet:**This option is available only if you provided a DNS label for the VCN during creation. If you want this subnet's instances to have DNS hostnames (which can be used with the built-in DNS capability in the VCN), select the check box for **Use DNS Hostnames in this Subnet**. Then you may specify a DNS label for the subnet, or the Console will generate one for you. The dialog box will automatically display the corresponding **DNS Domain Name** for the subnet (*<subnet DNS label>*.*<VCN DNS label>*.oraclevcn.com). For more information, see DNS in Your Virtual Cloud Network.
- **DHCP Options:** Select the default set of DHCP options.
- **Security Lists:** Select the default security list.

- **Tags:** Leave as is. You can add tags later if you want. For more information, see Resource Tags.

c. Click **Create Subnet**.

The subnet is then created and displayed on the **Subnets** page.

3. Update the default security list to include rules to allow the types of connections that your instances in the VCN will need:

a. While still on the page displaying your VCN's subnets, click **Security Lists**, and then click the default security list.

b. Under **Resources**, click either **Ingress Rules** or **Egress Rules** depending on the type of rule you want to work with. You can add one rule at a time by clicking either **Add Ingress Rule** or **Add Egress Rule**.

c. Add your desired rules. Here are suggested ones to add to the default ones already in the default security list:

Example: Ingress HTTP access

- **Type:** Ingress
- **Stateless**: Unselected (this is a stateful rule)
- **Source Type:** CIDR
- **Source CIDR:** 0.0.0.0/0
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range**: 80

Example: Ingress HTTPS access

- **Type:** Ingress
- **Stateless**: Unselected (this is a stateful rule)

- **Source Type:** CIDR
- **Source CIDR:** 0.0.0.0/0
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range**: 443

Example: Ingress SQL*Net access for Oracle databases

- **Type:** Ingress
- **Stateless**: Unselected (this is a stateful rule)
- **Source Type:** CIDR
- **Source CIDR:** 0.0.0.0/0
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range**: 1521

Example: Ingress RDP access required for Windows instances

- **Type:** Ingress
- **Stateless**: Unselected (this is a stateful rule)
- **Source Type:** CIDR
- **Source CIDR:** 0.0.0.0/0
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range**: 3389



Tip

For additional security, you could modify all the stateful ingress rules to allow traffic only from within your VCN and your on-premises network. You would need to create separate rules for each, one with the VCN's CIDR as the source, and one with the on-premises network's CIDR as the source.

For a production VCN, you typically set up one or more *custom* security lists for each subnet. You can edit the subnet to use different security lists if you like. If you choose not to use the default security list, do so only after carefully assessing which of its default rules you want to duplicate in your custom security list. For example: the default ICMP rules in the default security list are important for receiving connectivity messages.

Task 2: Create instances in separate availability domains

You can now create one or more instances in the subnet (see Launching an Instance). The scenario's diagram shows instances in two different availability domains. When you create the instance, you choose the AD, which VCN and subnet to use, and several other characteristics.

However, you can't yet communicate with the instances because there's no gateway connecting the VCN to your on-premises network. The next procedure walks you through creating an IPSec VPN connection to enable that communication.

Task 3: Add an IPSec VPN to your VCN

1. Create a customer-premises equipment (CPE) object:
    a. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Customer-Premises Equipment**.

b. Click **Create Customer-Premises Equipment**.

c. Enter the following:

- **Create in Compartment:** Leave the default value (the compartment you're currently working in).
- **Name:** A friendly name for the customer-premises equipment object. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
- **IP Address:** The public IP address of the CPE at your end of the VPN (see Prerequisites).

d. Click **Create**.

The CPE object will be in the "Provisioning" state for a short period.

2. Create a dynamic routing gateway (DRG):

a. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Dynamic Routing Gateways**.

b. Click **Create Dynamic Routing Gateway**.

c. For **Create in Compartment:** Leave the default value (the compartment you're currently working in).

d. Enter a friendly name for the DRG. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.

e. Click **Create**.

The DRG will be in the "Provisioning" state for a short period. Make sure it is done being provisioned before continuing.

3. Attach the DRG to your VCN:

a. Click the DRG that you just created.

b. On the left side of the page, click **Virtual Cloud Networks**.

c. Click **Attach to Virtual Cloud Network**.

d. Enter the following:

- **Virtual Cloud Network Compartment:** The compartment where the VCN resides.
- **Virtual Cloud Network:** The VCN you created earlier.
- **Associate with Route Table:** Do not select a route table here. You would associate a route table with a DRG attachment only if you're setting up an advanced scenario called transit routing.

e. Click **Attach**.

The attachment will be in the "Attaching" state for a short period before it's ready.

4. Update the default route table (which has no rules yet):

a. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.

b. Click your VCN.

c. Under **Resources**, click **Route Tables**, and then click the default route table.

d. Click **Add Route Rule**.

e. Enter the following:

- **Target Type:** Dynamic Routing Gateway. The VCN's attached DRG is automatically selected as the target, and you don't have to specify the target yourself.
- **Destination CIDR Block:** 0.0.0.0/0 (which means that all non-intra-VCN traffic that is not already covered by other rules in the route table will go to the target specified in this rule).

f. Click **Add Route Rule**.

The VCN's default route table now directs outbound traffic to the DRG and ultimately to your on-premises network.

5. Create an IPSec Connection:

a. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **IPSec Connections**.

b. Click **Create IPSec Connection**.

c. Enter the following:

- **Create in Compartment:** Leave the default value (the compartment you're currently working in).
- **Name:** Enter a friendly name for the IPSec connection. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
- **Customer-Premises Equipment Compartment:** Leave as is (the VCN's compartment).
- **Customer-Premises Equipment:** Select the CPE object you created earlier.
- **Dynamic Routing Gateway Compartment:** Leave as is (the VCN's compartment).
- **Dynamic Routing Gateway:** Select the DRG that you created earlier.
- **Static Route CIDR:** Enter at least one static route CIDR (see Prerequisites). If you need to add another, click **Add Static Route**. You can enter up to 10 static routes, and you can change the static routes later if you like.

d. Click **Show Advanced Options** and optionally provide the following items:

- **CPE IKE Identifier:** Oracle defaults to using the public IP address of the CPE. But if your CPE is behind a NAT device, you might need to enter a different value. You can either enter the new value here, or change the value later.
- **Tags:** Leave as is. You can add tags later if you want. For more information, see Resource Tags.

e. Click **Create IPSec Connection**.

The IPSec connection is created and displayed on the page. It will be in the Provisioning state for a short period.

The displayed tunnel information includes the IP address of the VPN headend and the tunnel's IPSec status (possible values are Up, Down, and Down for Maintenance). At this point, the status is Down. To view the tunnel's shared secret, click the Actions icon (three dots), and then click **View Shared Secret**.

f. Copy the Oracle VPN IP address and shared secret for each of the tunnels to an email or other location so you can deliver it to the network engineer who will configure the on-premises router.

For more information, see Configuring Your CPE. You can view this tunnel information here in the Console at any time.

You have now created all the components required for the IPSec VPN. But your network administrator must configure the CPE before network traffic can flow between your on-premises network and VCN.

Task 4: Configure your CPE

These instructions are for the network administrator.

1. Make sure you have the tunnel configuration information that Oracle provided during IPSec VPN setup. See Task 3: Add an IPSec VPN to your VCN.
2. Configure your CPE according to the information in Configuring Your CPE.

If there are already instances in the subnet, you can confirm the IPSec connection is up and running by connecting to the instances from your on-premises network.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operations:

1. CreateVcn: Make sure to include a DNS label for the VCN if you want the instances to have hostnames (see DNS in Your Virtual Cloud Network).
2. CreateSubnet: Create one regional private subnet. Include a DNS label for the subnet if you want the instances to have hostnames. Use the default route table, default security list, and default set of DHCP options.
3. CreateDrg: This creates a new dynamic routing gateway (DRG)
4. CreateDrgAttachment: This attaches the DRG to the VCN.
5. CreateCpe: Here you'll provide the public IP address of the CPE at your end of the VPN (see Prerequisites).
6. CreateIPSecConnection: Here you'll provide the static routes for your on-premises network (see Prerequisites). In return, you'll receive the configuration information that your network administrator needs in order to configure your CPE. If you need that information later, you can get it with GetIPSecConnectionDeviceConfig. For more information about the configuration, see Configuring Your CPE.
7. UpdateRouteTable: To enable communication via the VPN, update the default route table to include this route: a route rule with destination = 0.0.0.0/0, and destination target = the DRG you created earlier.
8. First call GetSecurityList to get the default security list, and then call UpdateSecurityList to add rules for the types of connections that your instances in the VCN will need. Be aware that `UpdateSecurityList` overwrites the entire set of rules. Here are some suggested rules to add:
    - Stateful ingress: Source type=CIDR, source CIDR=0.0.0.0/0, protocol=TCP, source port = all, destination port=80 (for HTTP).
    - Stateful ingress: Source type=CIDR, source CIDR=0.0.0.0/0, protocol=TCP, source port = all, destination port=443 (for HTTPS).
    - Stateful ingress: Source type=CIDR, source CIDR=0.0.0.0/0, protocol=TCP, source port = all, destination port=1521 (for SQL*Net access to Oracle databases).

- Stateful ingress: Source type=CIDR, source CIDR=0.0.0.0/0, protocol=TCP, source port=all, destination port=3389 (for RDP; required only if using Windows instances).



Tip

For additional security, you could modify all the stateful ingress rules to allow traffic only from within your VCN and your on-premises network. You would need to create separate rules for each, one with the VCN's CIDR as the source, and one with the on-premises network's CIDR as the source.

9. LaunchInstance: Create one or more instances in the subnet. The scenario's diagram shows instances in two different availability domains. When you create the instance, you choose the AD, which VCN and subnet to use, and several other characteristics. For more information, see Launching an Instance.



Important

Although you can create instances in the subnet, you won't be able to communicate with them from your on-premises network until your network administrator configures your CPE (see Configuring Your CPE). After that, your IPSec connection should be up and running. You can confirm its status by using GetIPSecConnectionDeviceStatus. You can also confirm the IPSec connection is up by connecting to the instances from your on-premises network.

Scenario C: Public and Private Subnets with a VPN

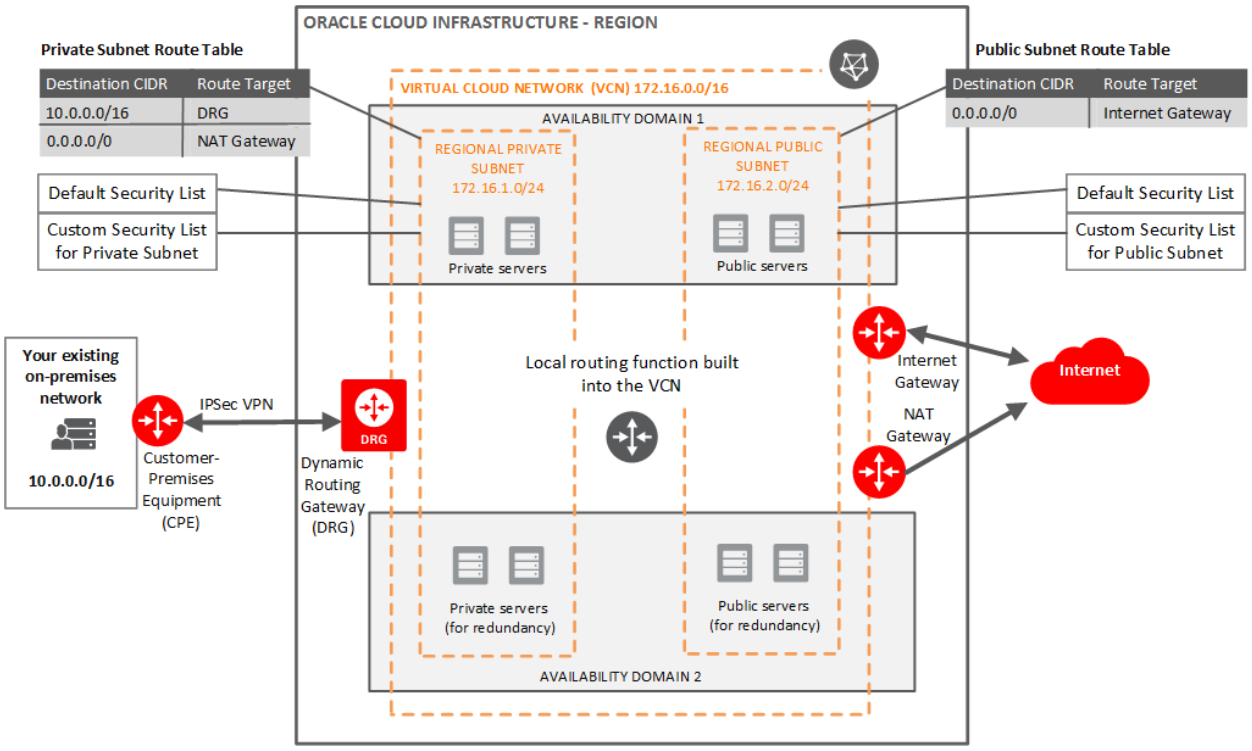
This topic explains how to set up Scenario C, which is a simple example of a multi-tier setup. It consists of a virtual cloud network (VCN) with a regional public subnet to hold public servers (such as web servers), and a regional private subnet to hold private servers (such as database servers). There are servers in separate availability domains for redundancy.

The VCN has a dynamic routing gateway (DRG) and IPSec VPN for connectivity to your on-premises network. Instances in the public subnet have direct access to the internet by way of an internet gateway. Instances in the private subnet can initiate connections to the internet by way of a NAT gateway (for example, to get software updates), but cannot receive inbound connections from the internet through that gateway.

Each subnet uses the default security list, which has default rules that are designed to make it easy to get started with Oracle Cloud Infrastructure. The rules enable typical required access (for example, inbound SSH connections and any type of outbound connections). Remember that security list rules only *allow* traffic. Any traffic not explicitly covered by a security list rule is denied.

Each subnet also has its own custom security list and custom route table with rules specific to the needs of the subnet's instances. In this scenario, the VCN's default route table (which is always empty to start with) is not used.

See the following figure.



Tip

The scenario uses an IPSec VPN for connectivity. However, you could instead use Oracle Cloud Infrastructure FastConnect.

Prerequisites

To set up the VPN in this scenario, you need to get the following information from a network administrator:

- IP address of the on-premises router at your end of the VPN
- Static routes for your on-premises network

You will provide Oracle this information and in return receive the information your network administrator needs in order to configure the on-premises router at your end of the VPN.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

If you're a member of the Administrators group, you already have the required access to execute Scenario C. Otherwise, you need access to Networking, and you need the ability to launch instances. See IAM Policies for Networking.

Setting Up Scenario C

Setup is easy in the Console. Alternatively, you can use the Oracle Cloud Infrastructure API, which lets you execute the individual operations yourself.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.



Important

Most of this process involves working with the Console or API (whichever you choose) for a short period to set up the desired Networking components. But there's also a critical step that requires a network administrator in your organization to take information you receive from setting up the components and use it to configure the on-premises router at your end of the VPN. Therefore you can't complete this process in one short session. You'll need to break for an unknown period of time while the network administrator completes the configuration and then return afterward to confirm communication with your instances over the VPN.

Using the Console

Task 1: Set up the VCN and subnets

1. Create the VCN:
    a. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
    b. Choose a compartment you have permission to work in (on the left side of the page). The page updates to display only the resources in that compartment. If you're not sure which compartment to use, contact an administrator. For more information, see Access Control.
    c. Click **Create Virtual Cloud Network**.
    d. Enter the following:

- **Name:** A friendly name for the VCN. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
- **Create in Compartment:** Leave as is.
- **Create Virtual Cloud Network Only:** Make sure this radio button is selected (the default).
- **CIDR Block:** A single, contiguous CIDR block for the VCN. For example: 172.16.0.0/16. You *cannot* change this value later. See Allowed VCN Size and Address Ranges. For reference, here's a CIDR calculator.
- **Use DNS Hostnames in this VCN:** If you want the instances in the VCN to have DNS hostnames (which can be used with the built-in DNS capability in the VCN), select the **Use DNS Hostnames in this VCN** check box. Then you can specify a DNS label for the VCN, or the Console will generate one for you. The dialog box automatically displays the corresponding **DNS Domain Name** for the VCN (*<VCN DNS label>*.oraclevcn.com). For more information, see DNS in Your Virtual Cloud Network.
- **Tags:** Leave as is. You can add tags later if you want. For more information, see Resource Tags.

e. Click **Create Virtual Cloud Network**.
The VCN is then created and displayed on the **Virtual Cloud Networks** page in the compartment you chose.

2. Create an internet gateway for your VCN:
   a. Under **Resources**, click **Internet Gateways**.
   b. Click **Create Internet Gateway**.
   c. Enter the following:
      - **Name:** A friendly name for the internet gateway. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.

- **Create in Compartment:** Leave as is.
- **Tags:** Leave as is. You can add tags later if you want. For more information, see Resource Tags.

d. Click **Create Internet Gateway**.
The internet gateway is then created and listed on the page.

3. Create a NAT gateway for your VCN:
   a. Under **Resources**, click **NAT Gateways**.
   b. Click **Create NAT Gateway**.
   c. Enter the following:
      - **Name:** A friendly name for the NAT gateway. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
      - **Create in Compartment:** Leave as is.
      - **Tags:** Leave as is. You can add tags later if you want. For more information, see Resource Tags.
   d. Click **Create NAT Gateway**.
   The NAT gateway is then created and listed on the page.
4. Create the custom route table for the public subnet (which you will create later):
   a. Under **Resources**, click **Route Tables**.
   b. Click **Create Route Table**.
   c. Enter the following:
      - **Name:** A friendly name for the route table (for example, *Public Subnet Route Table*). It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
      - **Create in Compartment:** Leave the default value (the compartment you're currently working in).

- Click **+ Additional Route Rule** and enter the following:
  - **Target Type:** Internet Gateway.
  - **Destination CIDR Block:** 0.0.0.0/0 (which means that all non-intra-VCN traffic that is not already covered by other rules in the route table will go to the target specified in this rule).
  - **Compartment:** Leave as is.
  - **Target:** The internet gateway you just created.

d. **Tags:** Leave as is. You can add tags later if you want. For more information, see Resource Tags.

e. Click **Create Route Table**.
   The route table is then created and listed on the page.

5. Create the custom route table for the private subnet (which you will create later):
   a. Click **Create Route Table**.
   b. Enter the following:
      - **Name:** A friendly name for the route table (for example, *Private Subnet Route Table*). It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
      - **Create in Compartment:** Leave the default value (the compartment you're currently working in).
      - Click **+ Additional Route Rule** and enter the following:
        - **Target Type:** NAT Gateway.
        - **Destination CIDR Block:** 0.0.0.0/0 (which means that all non-intra-VCN traffic that is not already covered by other rules in the route table will go to the target specified in this rule).
        - **Compartment:** Leave as is.
        - **Target:** The NAT gateway you just created.

c. **Tags:** Leave as is. You can add tags later if you want. For more information, see Resource Tags.

d. Click **Create Route Table**.

The route table is then created and listed on the page. Later on after you've set up the IPSec VPN, you will update the Private Subnet Route Table so it routes traffic from the private subnet to the on-premises network by way of the DRG.

6. Update the default security list to include rules to allow the types of connections that your instances in the VCN will need:

a. Under **Resources**, click **Security Lists**.

b. Click the default security list to view its details. By default, you land on the **Ingress Rules** page.

c. Edit each of the existing stateful ingress rules so that the **Source CIDR** is the CIDR for your on-premises network (10.0.0.0/16 in this example) and not 0.0.0.0/0. To edit an existing rule, click the Actions icon (three dots) for the rule, and then click **Edit**.

d. If you plan to launch Windows instances, add a rule to enable RDP access:

Example: Ingress RDP access required for Windows instances

- **Type:** Ingress
- **Stateless**: Unselected (this is a stateful rule)
- **Source Type:** CIDR
- **Source CIDR:** Your on-premises network (10.0.0.0/16 in this example)
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range**: 3389

7. Create a custom security list for the public subnet:

a. Return to the **Security Lists** page for the VCN.

b. Click **Create Security List**.

c. Enter the following:

- **Name:** Enter a friendly name for the list (for example, *Public Subnet Security List)*. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
- **Create in Compartment:** Leave the default value (the compartment you're currently working in).

d. Add the following ingress rules:

Example: Ingress HTTP access

- **Type:** Ingress
- **Stateless**: Unselected (this is a stateful rule)
- **Source Type:** CIDR
- **Source CIDR:** 0.0.0.0/0
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range**: 80

Example: Ingress HTTPS access

- **Type:** Ingress
- **Stateless**: Unselected (this is a stateful rule)
- **Source Type:** CIDR
- **Source CIDR:** 0.0.0.0/0

- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range**: 443

e. Add the following egress rule:

Example: Egress SQL*Net access to Oracle databases

- **Type:** Egress
- **Stateless**: Unselected (this is a stateful rule)
- **Destination Type:** CIDR
- **Destination CIDR:** CIDR for the private subnet (172.16.1.0/24 in this example)
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range**: 1521

f. Click **Create Security List**.
The custom security list for the public subnet is then created and listed on the page.

8. Create a custom security list for the *private* subnet:
   a. Click **Create Security List**.
   b. Enter the following:
      - **Name:** Enter a friendly name for the list (for example, *Private Subnet Security List)*. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
      - **Create in Compartment:** Leave the default value (the compartment you're currently working in).

c. Add the following ingress rules:

Example: Ingress SQL*Net access from clients in the public subnet

- **Type:** Ingress
- **Stateless**: Unselected (this is a stateful rule)
- **Source Type:** CIDR
- **Source CIDR:** CIDR for the public subnet (172.16.2.0/24 in this example)
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range**: 1521

Example: Ingress SQL*Net access from clients in the private subnet

- **Type:** Ingress
- **Stateless**: Unselected (this is a stateful rule)
- **Source Type:** CIDR
- **Source CIDR:** CIDR for the private subnet (172.16.2.1/24 in this example)
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range**: 1521

d. Add the following egress rules:

Example: Egress SQL*Net access to instances in the private subnet

- **Type:** Egress
- **Stateless**: Unselected (this is a stateful rule)

- **Destination Type:** CIDR
- **Destination CIDR:** CIDR for the private subnet (172.16.1.0/24 in this example)
- **IP Protocol:** TCP
- **Source Port Range:** All
- **Destination Port Range**: 1521

e. Click **Create Security List**.
The custom security list for the private subnet is then created and listed on the page.

9. Create the subnets in the VCN:
   a. Under **Resources**, click **Subnets**.
   b. Click **Create Subnet**.
   c. Enter the following:
      - **Name:** A friendly name for the regional public subnet (for example, *Regional Private Subnet*). It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
      - **Regional or Availability Domain-Specific:** Select **Regional** (recommended), which means the subnet spans all availability domains in the region. Later when you launch an instance, you can create it any availability domain in the region. For more information, see About Regional Subnets.
      - **CIDR Block:** A single, contiguous CIDR block within the VCN's CIDR block. For example: 172.16.1.0/24. You *cannot* change this value later. For reference, here's a CIDR calculator.
      - **Route Table:** Select the Private Subnet Route Table you created earlier.

- **Private or public subnet**: Select **Private Subnet**, which means VNICs in the subnet are not allowed to have public IP addresses. For more information, see Access to the Internet.
- **Use DNS Hostnames in this Subnet:**This option is available only if you provided a DNS label for the VCN during creation. If you want this subnet's instances to have DNS hostnames (which can be used with the built-in DNS capability in the VCN), select the check box for **Use DNS Hostnames in this Subnet**. Then you may specify a DNS label for the subnet, or the Console will generate one for you. The dialog box will automatically display the corresponding **DNS Domain Name** for the subnet (*<subnet DNS label>*.*<VCN DNS label>*.oraclevcn.com). For more information, see DNS in Your Virtual Cloud Network.
- **DHCP Options:** Select the default set of DHCP options.
- **Security Lists:** Select two security lists: Both the default security list and the Private Subnet Security List you created earlier.

d. Click **Create Subnet**.
   The private subnet is then created and displayed on the **Subnets** page.

e. Repeat the preceding steps a-d to create the *regional public subnet*. Instead use a name such as *Regional Public Subnet*, select **Public Subnet** instead of **Private Subnet**, use the Public Subnet Route Table, and use both the default security list and Public Subnet Security List you created earlier.

Task 2: Create instances in separate availability domains

You can now create one or more instances in the subnet (see Launching an Instance). The scenario's diagram shows instances in two different availability domains. When you create the instance, you choose the AD, which VCN and subnet to use, and several other characteristics.

For each instance in the public subnet, make sure to assign the instance a public IP address. Otherwise, you won't be able to reach the instance from your on-premises network.

You can't yet reach the instances in the private subnet because there's no gateway connecting the VCN to your on-premises network. The next procedure walks you through creating an IPSec VPN connection to enable that communication.

Task 3: Add an IPSec VPN to your VCN

1. Create a customer-premises equipment object:
    a. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Customer-Premises Equipment**.
    b. Click **Create Customer-Premises Equipment**.
    c. Enter the following:
        - **Create in Compartment:** Leave the default value (the compartment you're currently working in).
        - **Name:** A friendly name for the customer-premises equipment object. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
        - **IP Address:** The IP address of the on-premises router at your end of the VPN (see Prerequisites).
    d. Click **Create**.

    The CPE object is in the "Provisioning" state for a short period.
2. Create a dynamic routing gateway (DRG):
    a. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Dynamic Routing Gateways**.
    b. Click **Create Dynamic Routing Gateway**.
    c. For **Create in Compartment:** Leave the default value (the compartment you're currently working in).

d. Enter a friendly name for the DRG. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.

e. Click **Create**.

The DRG will be in the "Provisioning" state for a short period. Make sure it is done being provisioned before continuing.

3. Attach the DRG to your VCN:

   a. Click the DRG that you just created.

   b. On the left side of the page, click **Virtual Cloud Networks**.

   c. Click **Attach to Virtual Cloud Network**.

   d. Enter the following:

      - **Virtual Cloud Network Compartment:** The compartment where the VCN resides.
      - **Virtual Cloud Network:** The VCN you created earlier.
      - **Associate with Route Table:** Do not select a route table here. You would associate a route table with a DRG attachment only if you're setting up an advanced scenario called transit routing.

   e. Click **Attach**.

   The attachment will be in the "Attaching" state for a short period before it's ready.

4. Update the private subnet's route table (which already has one rule for the NAT gateway):

   a. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.

   b. Click your VCN.

   c. Click **Route Tables**, and then click the Private Subnet Route Table you created earlier.

   d. Click **Add Route Rule**.

e. Enter the following:

- **Target Type:** Dynamic Routing Gateway. The VCN's attached DRG is automatically selected as the target, and you don't have to specify the target yourself.
- **Destination CIDR Block:** 0.0.0.0/0 (which means that all non-intra-VCN traffic that is not already covered by other rules in the route table will go to the target specified in this rule).

f. Click **Add Route Rule**.

The table is updated to route any traffic destined for your on-premises network to the DRG. The original rule for 0.0.0.0/0 routes any remaining traffic leaving the subnet to the NAT gateway.

5. Create an IPSec Connection:

a. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **IPSec Connections**.

b. Click **Create IPSec Connection**.

c. Enter the following:

- **Create in Compartment:** Leave the default value (the compartment you're currently working in).
- **Name:** Enter a friendly name for the IPSec connection. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
- **Customer-Premises Equipment Compartment:** Leave as is (the VCN's compartment).
- **Customer-Premises Equipment:** Select the CPE object you created earlier.
- **Dynamic Routing Gateway Compartment:** Leave as is (the VCN's compartment).
- **Dynamic Routing Gateway:** Select the DRG that you created earlier.

- **Static Route CIDR:** Enter at least one static route CIDR (see Prerequisites). If you need to add another, click **Add Static Route**. You can enter up to 10 static routes, and you can change the static routes later if you like.

d. Click **Show Advanced Options** and optionally provide the following items:
   - **CPE IKE Identifier:** Oracle defaults to using the public IP address of the CPE. But if your CPE is behind a NAT device, you might need to enter a different value. You can either enter the new value here, or change the value later.
   - **Tags:** Leave as is. You can add tags later if you want. For more information, see Resource Tags.

e. Click **Create IPSec Connection**.
   The IPSec connection is created and displayed on the page. It will be in the Provisioning state for a short period.
   The displayed tunnel information includes the IP address of the VPN headend and the tunnel's IPSec status (possible values are Up, Down, and Down for Maintenance). At this point, the status is Down. To view the tunnel's shared secret, click the Actions icon (three dots), and then click **View Shared Secret**.

f. Copy the Oracle VPN IP address and shared secret for each of the tunnels to an email or other location so you can deliver it to the network engineer who will configure the on-premises router.
   For more information, see Configuring Your CPE. You can view this tunnel information here in the Console at any time.

You have now created all the components required for the IPSec VPN. But your network administrator must configure the on-premises router before network traffic can flow between your on-premises network and VCN.

Task 4: Configure your on-premises router

These instructions are for the network administrator.

1. Make sure you have the tunnel configuration information that Oracle provided during VPN setup. See Task 3: Add an IPSec VPN to your VCN.
2. Configure your on-premises router according to the information in Configuring Your CPE.

If there are already instances in one of the subnets, you can confirm the IPSec connection is up and running by connecting to the instances from your on-premises network. To connect to instances in the public subnet, you must connect to the instance's public IP address.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operations:

1. CreateVcn: Make sure to include a DNS label if you want the VCN to use the VCN Resolver (see DNS in Your Virtual Cloud Network).
2. CreateInternetGateway
3. CreateNatGateway
4. CreateRouteTable: Call it to create the Public Subnet Route Table. To enable communication by way of the internet gateway, add a route rule with destination = 0.0.0.0/0, and destination target = the internet gateway you created earlier.
5. CreateRouteTable: Call it again to create the Private Subnet Route Table. To enable communication by way of the NAT gateway, add a route rule with destination = 0.0.0.0/0, and destination target = the NAT gateway you created earlier.
6. First call GetSecurityList to get the default security list, and then call UpdateSecurityList:
    - Change the existing stateful ingress rules to use your on-premises network's CIDR as the source CIDR, instead of 0.0.0.0/0.

- If you plan to launch Windows instances, add this stateful ingress rule: Source type = CIDR, source CIDR = your on-premises network on TCP, source port = all, destination port = 3389 (for RDP).

7. CreateSecurityList: Call it to create the Public Subnet Security List with these rules:
   - Stateful ingress: Source type = CIDR, source 0.0.0.0/0 on TCP, source port = all, destination port = 80 (HTTP)
   - Stateful ingress: Source type = CIDR, source 0.0.0.0/0 on TCP, source port = all, destination port = 443 (HTTPS)
   - Stateful egress: Destination type = CIDR, destination CIDR blocks of private subnets on TCP, source port = all, destination port = 1521 (for Oracle databases)
8. CreateSecurityList: Call it again to create the Private Subnet Security List with these rules:
   - Stateful ingress: Source type = CIDR, source CIDR blocks of public subnets on TCP, source port = all, destination port = 1521 (for Oracle databases)
   - Stateful ingress: Source type = CIDR, source CIDR blocks of private subnets on TCP, source port = all, destination port = 1521 (for Oracle databases)
   - Stateful egress: Destination type = CIDR, destination CIDR blocks of private subnets on TCP, source port = all, destination port = 1521 (for Oracle databases)
9. CreateSubnet: Call it to create regional public subnet. Include a DNS label for the subnet if you want the VCN Resolver to resolve hostnames for VNICs in the subnet. Use the Public Subnet Route Table you created earlier. Use both the default security list and the Public Subnet Security List that you created earlier. Use the default set of DHCP options.
10. CreateSubnet: Call it again to create regional private subnet. Include a DNS label for the subnet if you want the VCN Resolver to resolve hostnames for VNICs in the subnet. Use the Private Subnet Route Table you created earlier. Use both the default security list and the Private Subnet Security List that you created earlier. Use the default set of DHCP options.
11. CreateDrg: This creates a new dynamic routing gateway (DRG).
12. CreateDrgAttachment: This attaches the DRG to the VCN.

13. CreateCpe: Here you provide the IP address of the router at your end of the VPN (see Prerequisites).
14. CreateIPSecConnection: Here you provide the static routes for your on-premises network (see Prerequisites). In return, you receive the configuration information your network administrator needs in order to configure your router. If you need that information later, you can get it with GetIPSecConnectionDeviceConfig. For more information about the configuration, see Configuring Your CPE.
15. First call GetRouteTable to get the Private Subnet Route Table. Then call UpdateRouteTable to add a route rule with destination = the on-premises network CIDR (10.0.0.0/16 in this example), and destination target = the DRG you created earlier.
16. LaunchInstance: Launch at least one instance in each subnet. By default, the instances in the public subnets are assigned public IP addresses. For more information, see Launching an Instance.

You can now communicate from your on-premises network with the instances in the public subnet over the internet gateway.



Important

Although you can launch instances into the private subnets, you can't communicate with them from your on-premises network until your network administrator configures your on-premises router (see Configuring Your CPE). After that, your IPSec connection should be up and running. You can confirm its status by using GetIPSecConnectionDeviceStatus. You can also confirm the IPSec connection is up by connecting to the instances from your on-premises network.

Advanced Scenario: Transit Routing

This topic explains an advanced networking scenario called *transit routing*. This scenario enables communication between an on-premises network and multiple virtual cloud networks (VCNs) over a single Oracle Cloud Infrastructure FastConnect or IPSec VPN.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Highlights

- You can use a single FastConnect or IPSec VPN to connect your on-premises network with *multiple* VCNs in the same region, in a *hub-and-spoke* layout.
- The VCNs must be in the same region but can be in different tenancies. For accurate routing, the CIDR blocks of the various subnets of interest in the on-premises network and VCNs must not overlap.
- The *hub* VCN uses a dynamic routing gateway (DRG) to communicate with the on-premises network. The hub VCN peers with each *spoke* VCN. The hub and spoke VCNs use local peering gateways (LPGs) to communicate.
- To enable the desired traffic from the on-premises network through the hub VCN to a peered spoke VCN, you implement route rules for the hub VCN's DRG attachment and LPG, and for the spoke VCN's subnets.
- By configuring route tables that reside in the hub VCN, you can control whether a particular subnet in a peered spoke VCN is advertised to the on-premises network, and whether a particular subnet in the on-premises network is advertised to a peered spoke VCN.

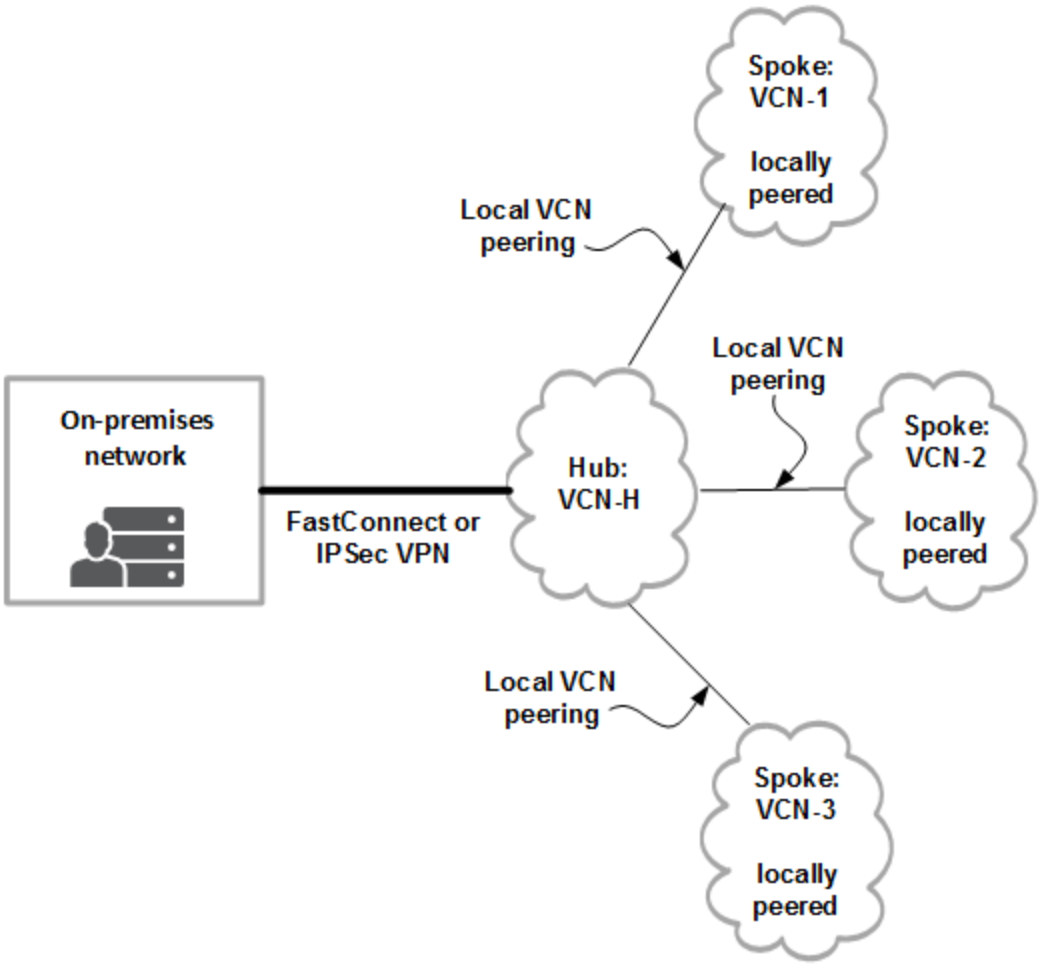
Overview of Transit Routing

A basic networking scenario involves connecting your on-premises network to a VCN with either Oracle Cloud Infrastructure FastConnect or an IPSec VPN. These two basic scenarios illustrate that layout: Scenario B: Private Subnet with a VPN and Scenario C: Public and Private Subnets with a VPN.

There's an advanced networking scenario that lets you use your single FastConnect or IPSec VPN to communication with *multiple VCNs* from your on-premises network. The VCNs must be in the same region but can be in different tenancies.

Here's a basic example of why you might use transit routing: you have a large organization with different departments, each with their own VCN. Your on-premises network needs access to the different VCNs, but you don't want the administration overhead of maintaining a secure connection from each VCN to the on-premises network. Instead you want to use a single FastConnect or IPSec VPN.

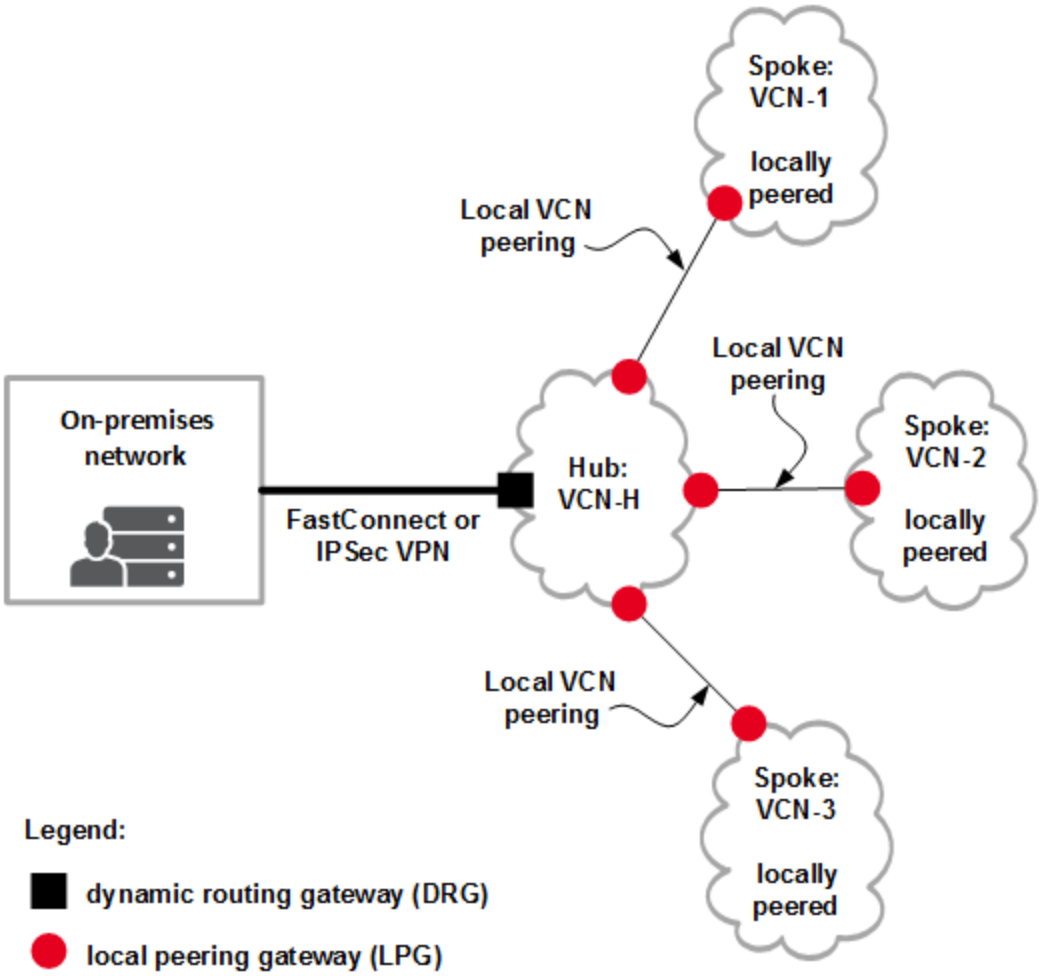
The scenario uses a "hub and spoke" layout, as illustrated in the following diagram.



One of the VCNs acts as the hub (VCN-H) and connects to your on-premises network by way of FastConnect or an IPSec VPN. The other VCNs are locally peered with the hub VCN. The traffic between the on-premises network and the peered VCNs transits through the hub VCN. The VCNs must be in the same region but can be in different tenancies.

Gateways Involved in Transit Routing

The next diagram shows the gateways on the VCNs. The hub VCN has a dynamic routing gateway (DRG), which is the communication path with the on-premises network. For each locally peered spoke VCN, there's a pair of local peering gateways (LPGs) that anchor the peering connection. One LPG is on the hub VCN, and the other is on the spoke VCN.





Tip

If you're already familiar with the Networking service and local VCN peering, these are the most important new concepts to understand:

- For each spoke VCN subnet that needs to communicate with the on-premises network, you must update the subnet's route table with a rule that sets the target (the next hop) as the spoke VCN's LPG for all traffic destined for the on-premises network.
- You must add a route table to the hub VCN, associate it with *the DRG attachment*, and add a route rule that sets the target (the next hop) as the hub VCN's LPG (for that spoke) for all traffic destined for that spoke VCN (or a specific subnet in that VCN).
- You must add another route table to the hub VCN, associate it with *the hub VCN's LPG* (for that spoke), and add a route rule that sets the target (the next hop) as the DRG for all traffic destined for the on-premises network (or a specific subnet in that network).

See the instructions in Task 5: Add a route rule to the spoke VCN's subnet and Task 6: Set up ingress routing between the DRG and LPG on the hub VCN.

Example: Components and Routing for a Hub and Single Spoke

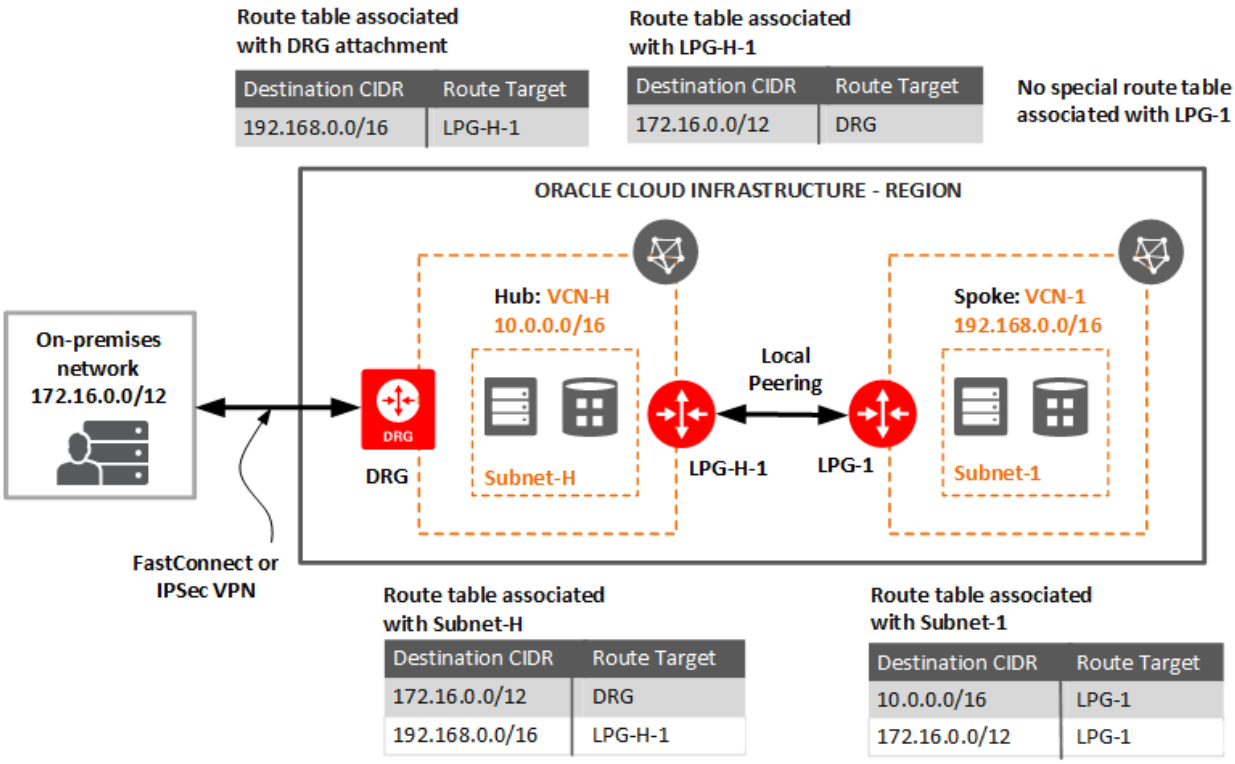
This example includes a hub VCN and only a single spoke VCN for simplicity. Each VCN also has a single subnet for simplicity. Notice that to make transit routing work, you do not have to create a subnet in the hub VCN. However, you can if you like. The subnet can contain cloud resources that your on-premises network or the spoke VCN need to use.

The following diagram shows the basic layout for this example.



Note

In a hub-and-spoke model, the hub VCN can have multiple spokes and therefore multiple LPGs (one per spoke). This topic uses the phrase *the hub VCN's LPG*, which could therefore be ambiguous. When the phrase is used here, it means the hub LPG for the *particular spoke of interest.* In the following diagram, it's LPG-H-1. Additional spokes would involve creation of an LPG-H-2, LPG-H-3, and so on.



The diagram also shows the required Networking service route tables and route rules for transit routing through the hub VCN. The diagram has four route tables, each associated with a different resource:

- **DRG attachment:**
  - The route table belongs to the hub VCN and is associated with the DRG *attachment*. Why the attachment and not the DRG itself? Because the DRG is a standalone resource that you can attach to any VCN in the same region and tenancy as the DRG. The attachment itself identifies which VCN.
  - The route table routes the inbound traffic that is from the on-premises network and destined for the spoke VCN (VCN-1). The rule sends that traffic to LPG-H-1.
- **LPG-H-1:**

    - This route table belongs to the hub VCN and is associated with LPG-H-1.
    - The route table routes inbound traffic that is from VCN-1 and destined for the on-premises network. The rule sends that traffic to the DRG.
- **Subnet-H**
    - This route table belongs to the hub VCN and is associated with subnet-H.
    - It includes rules to enable traffic with the on-premises network and with subnet-1.
- **Subnet-1**
    - This route table belongs to the spoke VCN and is associated with subnet-1.
    - It includes rules to enable traffic with subnet-H and the on-premises network.

Here are some additional important details to note:

- A route table that is associated with a DRG attachment can have only rules that target an LPG. Conversely, a route table that is associated with an LPG can have only rules that target a DRG. These rules route the traffic *through* the hub VCN and to the spoke VCN or on-premises network.
- Even though the preceding statement is true, inbound traffic to subnets *within the hub VCN* is still allowed. You do not need to set up explicit rules for this inbound traffic in the DRG attachment's route table or hub LPG's route table. When this kind of inbound traffic reaches the DRG or the hub LPG, the traffic is automatically routed to its destination in the hub VCN. And in general: for any route table belonging to a given VCN, you can't create a rule that lists that VCN's CIDR (or a sub-section) as the rule's destination.
- A DRG attachment can exist without a route table associated with it. However, after you associate a route table with a DRG attachment, there must always be a route table associated with it. But, you can associate a *different* route table. You can also edit the table's rules, or delete some or all of the rules.

About CIDR Overlap

In this example, the various networks do not have overlapping CIDR blocks (172.16.0.0/12 versus 10.0.0.0/16 versus 192.168.0.0/16). The Networking service does not allow local VCN

peering between two VCNs with overlapping CIDRs. That means each spoke must not overlap with the hub.

However, the Networking service does not validate whether the spoke VCNs themselves overlap with each other, or if any of the VCNs overlap with the on-premises network. You must ensure that CIDRs for all the subnets that need to communicate with each other don't overlap. Otherwise, traffic may be dropped.

A Networking service route table cannot contain two rules with the exact same destination CIDR. However, if two rules in the same route table have overlapping destination CIDRs, the most specific rule in the table is used to route the traffic (that is, the rule with the longest prefix match).

Route Advertisement to the On-Premises Network and Spoke VCNs

From a security standpoint, you can control route advertisement so that only specific subnets in the on-premises network are advertised to the spoke VCNs. Similarly, you can control which subnets in the spoke VCNs are advertised to the on-premises network.

The routes advertised to the on-premises network consist of:

- The rules listed in the route table associated with the DRG attachment (192.168.0.0/16 in the preceding diagram)
- The individual subnets in the hub VCN

The routes advertised to the spoke VCN consist of:

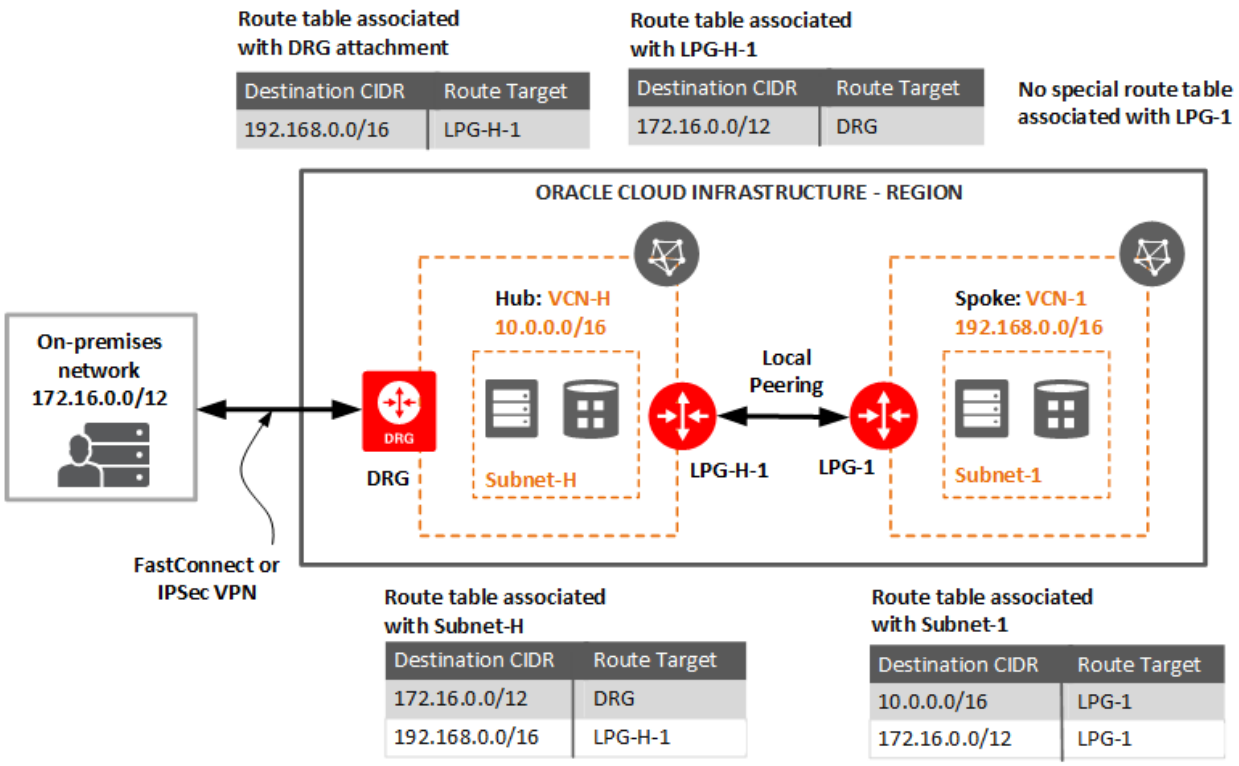
- The individual subnets in the hub VCN
- The rules listed in the route table associated with the hub VCN's LPG for the spoke (172.16.0.0/12 in the preceding diagram)

Therefore, the administrator *of the hub VCN* alone can control which routes are advertised to the on-premises network and spoke VCNs.

In the preceding example, the relevant routes use the full CIDR block of the on-premises network and spoke VCN as the destination (172.16.0.0/12 and 192.168.0.0/16, respectively), but they could instead use a subnet of those networks to restrict routing to specific subnets.

Details About Routing for Different Traffic Paths

To further illustrate how routing takes place in the preceding example, let's look more closely at different paths of traffic. Here's the same diagram again:



Traffic from the on-premises network to the spoke VCN

1. Traffic leaves the on-premises network and reaches the DRG. The traffic's destination is in subnet-1 (for example, 192.168.0.5).
2. The DRG attachment's associated route table has a rule for 192.168.0.0/16. It matches the destination and sends the traffic to the rule's target, LPG-H-1. Remember that you

can use the rules in this route table to control which subnets in the spoke VCN are advertised to the on-premises network. You could instead set up the rule to list only a subnet of the spoke VCN.

3. LPG-H-1 receives the traffic.
4. Egress traffic leaving a VCN through an LPG is automatically routed to the LPG's peered LPG, which is LPG-1 in this situation. That routing occurs automatically because of the peering connection between the two LPGs.
5. LPG-1 receives the traffic.
6. Traffic coming in to a VCN through an LPG is automatically routed to the destination within the VCN. No explicit route rules are required.

Traffic from the spoke VCN to the on-premises network

1. Traffic comes from an instance in subnet-1 in the spoke VCN. The traffic's destination is in the on-premises network (for example, 172.16.0.3).
2. Subnet-1's associated route table has a rule for 172.16.0.0/12. It matches the destination and sends the traffic to the rule's target, LPG-1.
3. LPG-1 receives the traffic.
4. Egress traffic leaving a VCN through an LPG is automatically routed to the LPG's peered LPG, which is LPG-H-1 in this situation. That routing occurs automatically because of the peering connection between the two LPGs.
5. LPG-H-1 receives the traffic.
6. LPG-H-1's associated route table has a rule for 172.16.0.0/12. It matches the destination and sends the traffic to the rule's target, the DRG. Remember that you can use the rules in this route table to control which subnets in the on-premises network are advertised to the spoke VCN. You could instead set up the rule to list only a subnet of the on-premises network.
7. The DRG receives the traffic.

8. Egress traffic leaving the VCN through the DRG is routed based on the IPSec VPN and FastConnect configuration. No explicit rules in the DRG attachment's route table are required.

Notice that subnet-1 in the spoke VCN and LPG-H-1 both have route rules with 172.16.0.0/12 as the destination CIDR. Those rules don't have to use the exact same CIDR block. However, make sure both rules cover the traffic you want to route from the spoke to the on-premises network. The rule in subnet-1's route table controls which traffic is routed from subnet-1 to LPG-H-1. The rule in LPG-H-1's route table controls which traffic is routed from the spoke VCN to the on-premises network. If LPG-H-1's route rule is more restrictive than subnet-1's route rule, some traffic leaving the subnet could ultimately be dropped and not reach the DRG.

Traffic from the spoke VCN to a subnet in the hub VCN

Depending on your situation, you might want to enable traffic between instances in the two VCNs, and not just traffic between the on-premises VCN and the spoke VCN. Here's how traffic would flow from the spoke VCN to a destination in the hub VCN:

1. Traffic comes from an instance in subnet-1 in the spoke VCN. The traffic's destination is in a subnet in the hub VCN (for example, 10.0.0.3).
2. Subnet-1's associated route table has a rule for 10.0.0.0/16. It matches the destination and sends the traffic to the rule's target, LPG-1.
3. LPG-1 receives the traffic.
4. Egress traffic leaving a VCN through an LPG is automatically routed to the LPG's peered LPG, which is LPG-H-1 in this situation. That routing occurs automatically because of the peering connection between the two LPGs.
5. LPG-H-1 receives the traffic.
6. Traffic coming in to a VCN through an LPG is automatically routed to the destination within the VCN. No explicit route rules are required.

A similar series of routing steps occurs for traffic going from subnet-H to subnet-1, but in the reverse direction. Subnet-H's route table has a rule that matches the spoke VCN's CIDR (192.168.0.0/16) and sends the traffic to LPG-H-1, which forwards it on to LPG-1.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

If you're a member of the Administrators group, you already have the required access to set up transit routing. Otherwise, you need access to the Networking service, and you need the ability to launch instances. See IAM Policies for Networking.

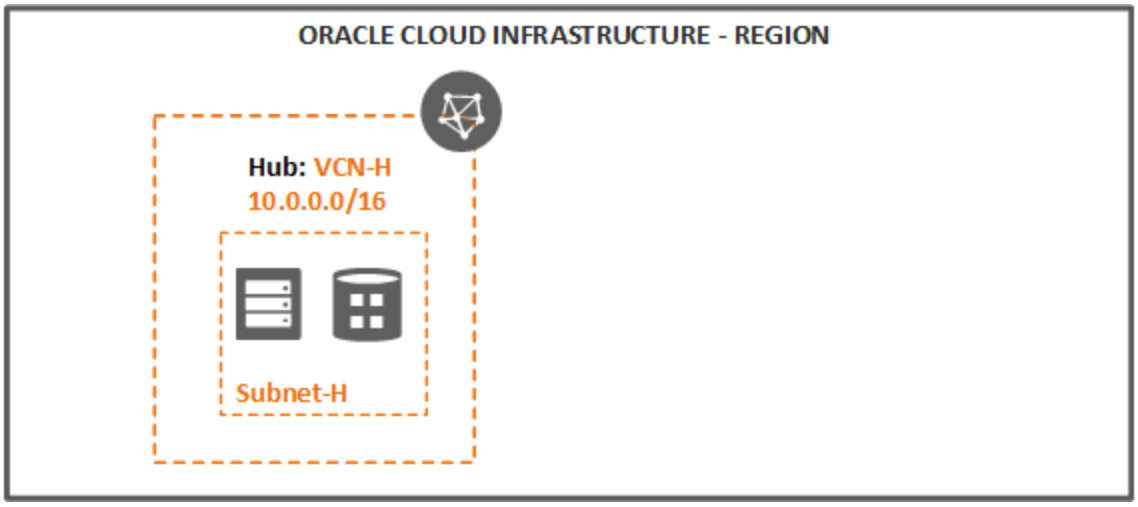
Setting Up VCN Transit Routing in the Console



Tip

You might already have many of the necessary Networking components and connections in this advanced scenario already set up. So you might be able to skip some of the following tasks. **If you already have a hub VCN connected to your on-premises network, and spoke VCNs locally peered with the hub VCN, then Task 5 and Task 6 are the most important.** They enable traffic to be routed between your on-premises network and the spoke VCN.

Task 1: Set up the hub VCN

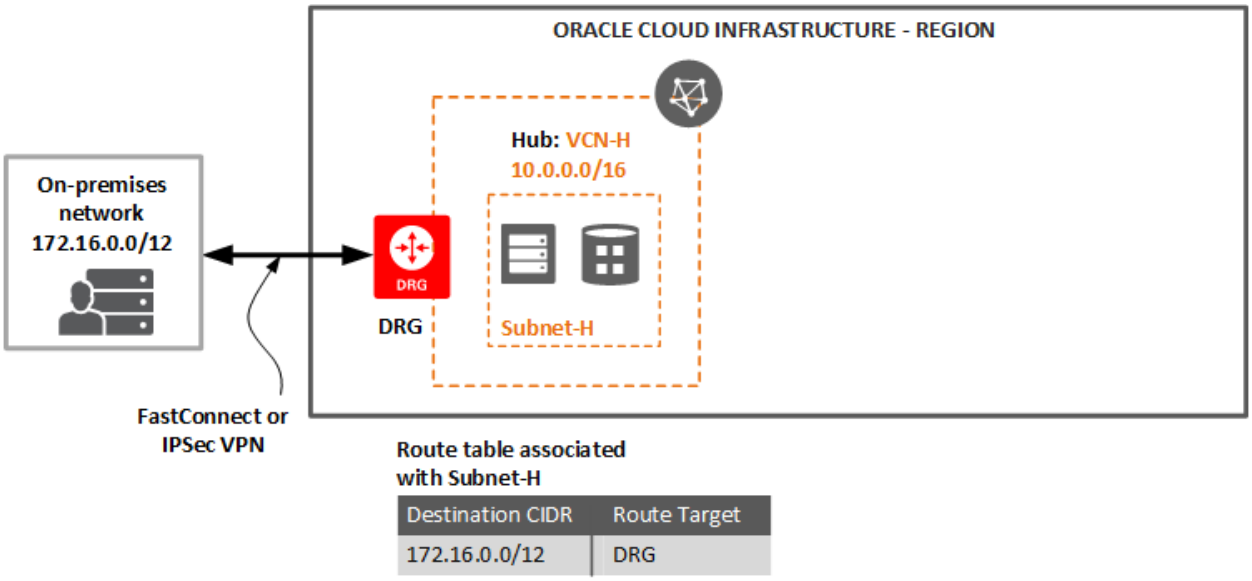


In this task, you set up the hub VCN. To make transit routing work, you do not have to create a subnet in the hub VCN. However, this example includes one. The subnet can contain cloud resources that your on-premises network or the spoke VCN need to use.

For more information and instructions:

- VCNs and Subnets

Task 2: Connect the hub VCN with your on-premises network

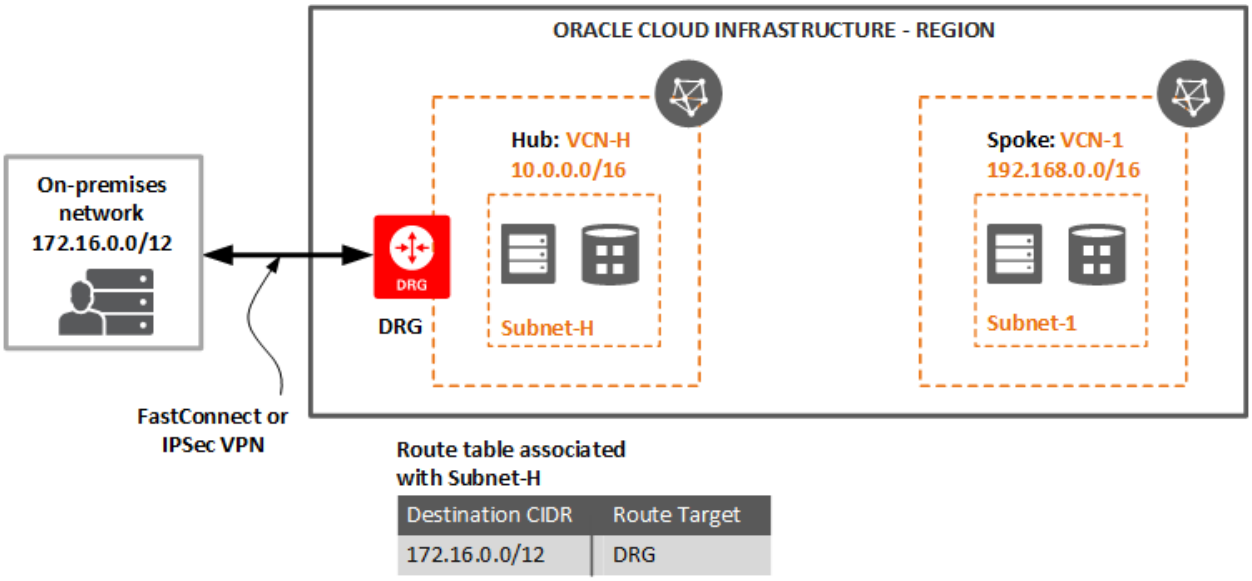


In this task, you set up either FastConnect or an IPSec VPN between your hub VCN and your on-premises network. As part of this process, you attach a DRG to the hub VCN and set up routing between the hub VCN and your on-premises network.

Notice that you do not yet create the route table that will be associated with the DRG attachment. That comes in a later step.

- FastConnect
- VPN Connect
- Dynamic Routing Gateways (DRGs)

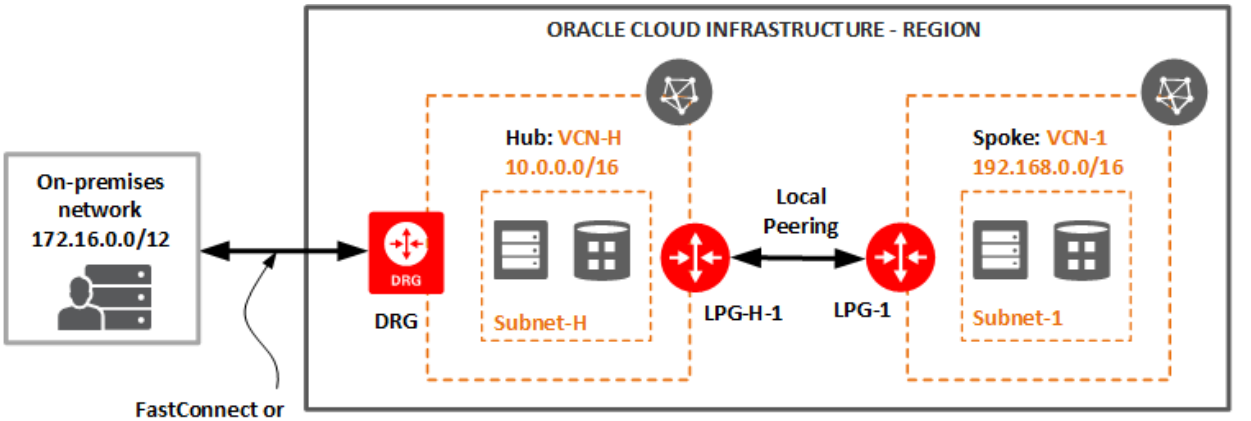
Task 3: Set up a spoke VCN with at least one subnet



In this task, you set up the spoke VCN with at least one subnet. For more information and instructions:

- VCNs and Subnets

Task 4: Set up a local peering between the hub VCN and the spoke VCN



Route table associated with Subnet-H

Destination CIDR	Route Target
172.16.0.0/12	DRG
192.168.0.0/16	LPG-H-1

Route table associated with Subnet-1

Destination CIDR	Route Target
10.0.0.0/16	LPG-1

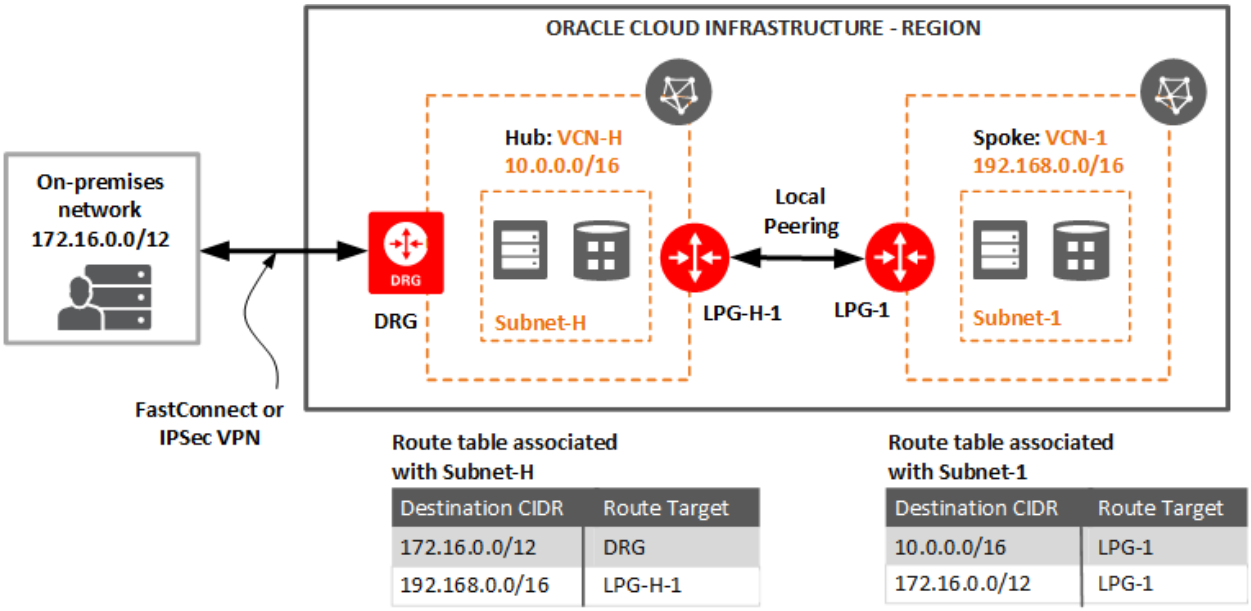
In this task, you add an LPG to each VCN, establish a connection between the LPGs, and set up routing that enables resources in one VCN to communicate with resources in the other.

Notice that you do not yet create the route table that will be associated with the LPG on the hub VCN (LPG-H-1). That comes in a later step.

For more information and instructions:

- Setting Up a Local Peering

Task 5: Add a route rule to the spoke VCN's subnet



In this task, you add a rule to the route table associated with the spoke VCN's subnet. This rule routes traffic that is destined for the on-premises network to the spoke VCN's LPG (LPG-1 in the diagram).

Prerequisites: You already have an LPG for the spoke VCN, and a route table associated with the subnet (on the spoke VCN) that needs to communicate with the on-premises network.

1. For the spoke VCN, view the list of subnets.
2. For the subnet of interest, look at its details and click the link for its associated route table.
3. Edit the route table to include a rule that sends traffic to the on-premises network:

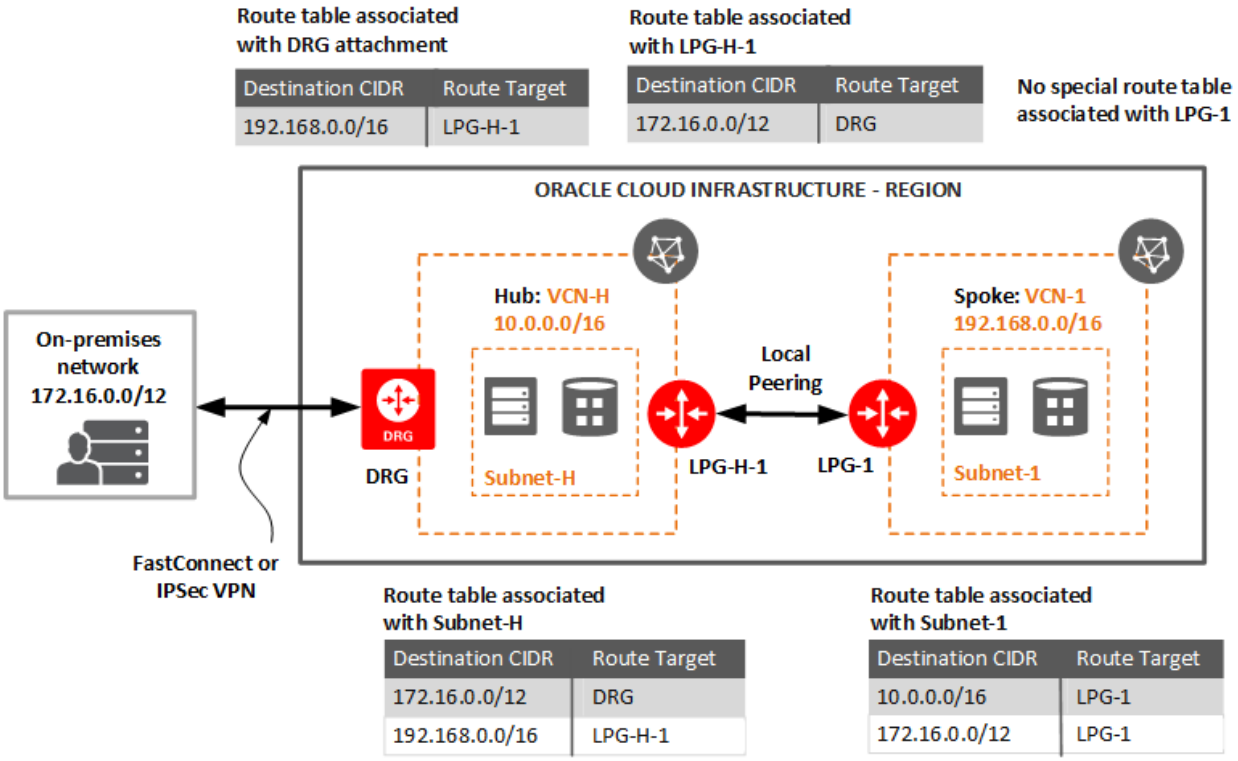
a. Click **Add Route Rule**.

b. Enter this information for the route rule:

- **Target Type:** Local Peering Gateway.
- **Destination CIDR Block**: The on-premises network's CIDR (172.16.0.0/12 in the earlier example).
- **Compartment:** The compartment where the spoke VCN's LPG is located.
- **Target:** The spoke VCN's LPG.

c. Click **Add Route Rule**.

Task 6: Set up ingress routing between the DRG and LPG on the hub VCN



In this task, you set up the route tables for the DRG attachment and hub VCN's LPG for the spoke of interest (LPG-H-1).

Prerequisites:

- You already have a DRG attached to the hub VCN.
- You already have a hub VCN LPG for the spoke of interest.

1. Create a route table for the DRG attachment:
    a. In the Console, view the hub VCN's details.
    b. Under **Resources**, click **Route Tables** to view the VCN's route tables.
    c. Click **Create Route Table**.
    d. Enter the following:
        - **Name:** A descriptive name for the route table. Example: DRG Route Table. Avoid entering confidential information.
        - **Create in Compartment:** Leave as is.
    e. Click **+ Additional Route Rule**, and enter this information for the route rule:
        - **Target Type:** Local Peering Gateway.
        - **Destination CIDR Block**: This spoke VCN's CIDR (192.168.0.0/16 in the earlier example). Remember that you can use the routes in this table to control which subnets in the spoke VCN are advertised to the on-premises network. You could instead set up the rule to list only a particular subnet of the spoke VCN that the on-premises network.
        - **Compartment:** The compartment where the hub VCN's LPG is located.
        - **Target:** The hub VCN's LPG.
    f. Click **Create Route Table**.
    The route table is created and displayed in the list.
2. Associate the route table you just created (called *DRG Route Table* in this example) with the hub VCN's DRG attachment:
    a. While still viewing the hub VCN's details, click **Dynamic Routing Gateways** to view the attached DRG.
    b. Click the Actions icon (three dots), and then click **Associate With Route Table**.
    c. Enter the following:
        - **Route Table Compartment:** Select the compartment of the route table you created in the previous step.

- **Route Table:** Select the route table you created in the previous step.

d. Click **Associate**.
The route table is associated with the DRG attachment.

3. Create a special route table for the hub VCN's LPG for this spoke:
   a. While still viewing the hub VCN's details, click **Route Tables**.
   b. Click **Create Route Table**.
   c. Enter the following:
      - **Create in Compartment:** Leave as is.
      - **Name:** A descriptive name for the route table. Recommendation: `Hub LPG-# Route Table` (where # indicates which spoke). Avoid entering confidential information.
   d. Click **+ Additional Route Rule**, and enter this information for the route rule:
      - **Target Type:** Dynamic Routing Gateway. The VCN's attached DRG is automatically selected as the target, and you don't have to specify the target yourself.
      - **Destination CIDR Block**: The on-premises network's CIDR (172.16.0.0/12 in the earlier example). Remember that you can use the routes in this table to control which subnets in the on-premises network are advertised to this spoke VCN. You could instead set up the rule to list only a subnet of the on-premises network that needs to communicate with this spoke.
   e. Click **Create Route Table**.
   The route table is created and displayed in the list.
4. Associate the route table you created in the previous step (called *Hub LPG-# Route Table* in this example) with the hub VCN's LPG for the spoke of interest:
   a. While still viewing the hub VCN's details, click **Local Peering Gateways** to view the hub VCN's LPG for this spoke.

b. For the LPG you're interested in, click the Actions icon (three dots), and then click **Associate With Route Table**.

c. Enter the following:

- **Route Table Compartment:** Select the compartment of the route table you created in the previous step.
- **Route Table:** Select the route table you created in the previous step.

d. Click **Associate**.
The route table is associated with the LPG.

If you need more spoke VCNs, here is the general process for each spoke VCN:

1. Repeat Tasks 3-5 for the new spoke VCN.
2. Repeat Task 6 with these changes:
   - For Step 1: Instead of creating a new route table for the DRG attachment, update the existing route table to include a new rule for the new spoke VCN. The destination CIDR is the spoke VCN's CIDR (or a subnet within). The target is the hub VCN's LPG for the new spoke.
   - For Step 2: Skip this step entirely because the DRG attachment is already associated with its route table.
   - For Step 3: Repeat as is. Name the new route table according to which spoke the route table is for (for example, `Hub LPG-2 Route Table` for the second spoke).
   - For Step 4: Repeat as is. Associate the new route table you created in Step 3 with the hub VCN's LPG for the new spoke.

Turning Off Transit Routing

To turn off transit routing, remove the rules from:

- The route table associated with the DRG attachment.
- The route table associated with each LPG on the hub VCN.

A route table can be associated with a resource but have no rules. Without at least one rule, a route table does nothing.

A DRG attachment or LPG can exist without a route table associated with it. However, after you associate a route table with a DRG attachment or LPG, there must always be a route table associated with it. But, you can associate a *different* route table. You can also edit the table's rules, or delete some or all of the rules.

Changes to the API

For information about changes to the Networking service API to support transit routing, see the transit routing release notes.

VCNs and Subnets

This topic describes how to manage virtual cloud networks (VCNs) and the subnets in them. This topic uses the terms *virtual cloud network*, *VCN*, and *cloud network* interchangeably. The Console uses the term *Virtual Cloud Network*, whereas for brevity the API uses *VCN*.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Overview of VCNs and Subnets

A VCN is a software-defined network that you set up in the Oracle Cloud Infrastructure data centers in a particular region. A subnet is a subdivision of a VCN. For an overview of VCNs, allowed size, default VCN components, and scenarios for using a VCN, see Overview of Networking.

You can privately connect a VCN to another VCN so that the traffic does not traverse the internet. The CIDRs for the two VCNs must not overlap. For more information, see Access to Other VCNs: Peering. For an example of an advanced routing scenario that involves the peering of multiple VCNs, see Advanced Scenario: Transit Routing.

Each subnet in a VCN consists of a contiguous range of IP addresses that do not overlap with other subnets in the VCN. Example: 172.16.1.0/24. The first two IP addresses and the last in the subnet's CIDR are reserved by the Networking service. You can't change the size of the subnet after creation, so it's important to think about the size of subnets you need before creating them.

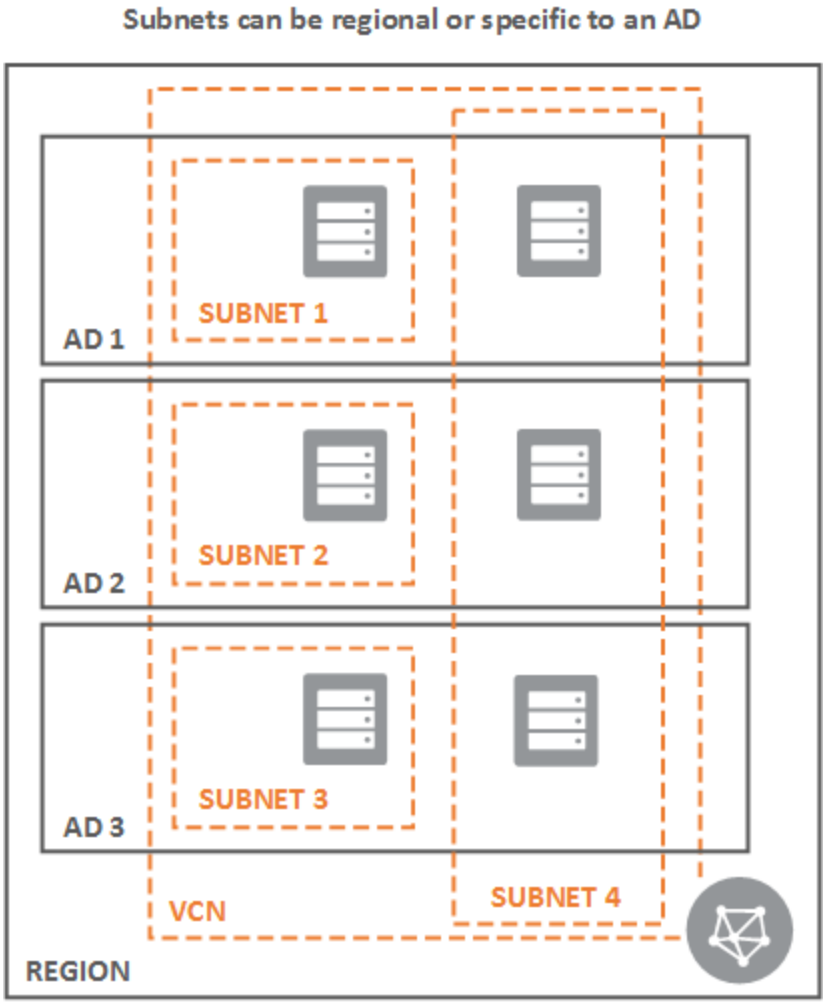
Subnets act as a unit of configuration: all instances in a given subnet use the same route table, security lists, and DHCP options. For more information, see Default Components that Come With Your VCN.

Subnets can be either public or private (see Public vs. Private Subnets). You choose this during subnet creation, and you can't change it later.

You can think of each Compute instance as residing in a subnet. But to be precise, each instance is actually attached to a virtual network interface card (VNIC), which in turn resides in the subnet and enables a network connection for that instance.

About Regional Subnets

Originally subnets were designed to cover only one availability domain (AD) in a region. They were all *AD-specific*, which means the subnet's resources were required to reside in a particular availability domain. Now subnets can be either AD-specific or *regional*. You choose the type when you create the subnet. Both types of subnets can co-exist in the same VCN. In the following diagram, subnets 1-3 are AD-specific, and subnet 4 is regional.



Aside from the removal of the AD constraint, regional subnets behave the same as AD-specific subnets. **Oracle recommends using regional subnets** because they're more flexible. They make it easier to efficiently divide your VCN into subnets while also designing for availability domain failure.

When you create a resource such as a Compute instance, you choose which availability domain the resource will be in. From a virtual networking standpoint, you must also choose

which VCN and subnet the instance will be in. You can either choose a regional subnet, or choose an AD-specific subnet that matches the AD you chose for the instance.



Warning

If anyone in your organization implements a regional subnet, be aware that you **may need to update any client code that works with Networking service subnets and private IPs**. There are possible breaking API changes. For more information, see the regional subnet release note.

Working with VCNs and Subnets

One of the first things you do when working with Oracle Cloud Infrastructure resources is create a VCN with one or more subnets. You can easily get started in the Console with a simple VCN and some related resources that enable you to launch and connect to an instance. See Tutorial - Launching Your First Linux Instance or Tutorial - Launching Your First Windows Instance.

For the purposes of access control, when you create a VCN or subnet, you must specify the compartment where you want the resource to reside. Consult an administrator in your organization if you're not sure which compartment to use. For more information, see Access Control.

You may optionally assign friendly names to the VCN and its subnets. The names don't have to be unique, and you can change them later. Oracle automatically assigns each resource a unique identifier called an Oracle Cloud ID (OCID). For more information, see Resource Identifiers.

You can also add a DNS label for the VCN and each subnet, which are required if you want the instances to use the *Internet and VCN Resolver* feature for DNS in the VCN. For more information, see DNS in Your Virtual Cloud Network.

When you create a subnet, you may optionally specify a route table for the subnet to use. If you don't, the subnet uses the cloud network's default route table. You can change which route table the subnet uses at any time.

Also, you may optionally specify one or more security lists for the subnet to use (up to five). If you don't specify any, the subnet uses the cloud network's default security list. You can change which security list the subnet uses at any time. Remember that the security list rules are enforced at the instance level, even though the list is associated at the subnet level.

Similarly, you may optionally specify a set of DHCP options for the subnet to use. All instances in the subnet will receive the configuration specified in that set of DHCP options. If you don't specify a set, the subnet uses the cloud network's default set of DHCP options. You can change which set of DHCP options the subnet uses at any time.

To delete a subnet, it must contain no resources (no instances, load balancers, DB systems, and orphaned mount targets). For more details, see Subnet or VCN Deletion.

To delete a VCN, its subnets must contain no resources. Also, the VCN must have no attached gateways. If you're using the Console, there's a "Delete All" process you can use after first ensuring the subnets are empty. See To delete a VCN.

For information about the number of VCNs and subnets you can have, see Service Limits.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: see IAM Policies for Networking.

Using the Console

To create a VCN



Note

The following procedure creates a VCN without any subnets or gateways for access. You must manually create the subnets and other resources before you can use the VCN. For a quick procedure that creates a VCN that you can try out immediately (that is, with subnets and an internet gateway), see Scenario A: Public Subnet.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Choose a compartment you have permission to work in (on the left side of the page). The page updates to display only the resources in that compartment. If you're not sure which compartment to use, contact an administrator. For more information, see Access Control.
3. Click **Create Virtual Cloud Network**.
4. Enter the following:
    a. **Name:** A friendly name for the VCN. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
    b. **Create in Compartment:** Leave as is.
    c. **Create Virtual Cloud Network Only:** Make sure this radio button is selected (the default).

d. **CIDR Block:** A single, contiguous CIDR block for the VCN. For example: 172.16.0.0/16. You *cannot* change this value later. See Allowed VCN Size and Address Ranges. For reference, here's a CIDR calculator.

e. **Use DNS Hostnames in this VCN:** If you want the instances in the VCN to have DNS hostnames (which can be used with the built-in DNS capability in the VCN), select the **Use DNS Hostnames in this VCN** check box. Then you can specify a DNS label for the VCN, or the Console will generate one for you. The dialog box automatically displays the corresponding **DNS Domain Name** for the VCN (*<VCN DNS label>*.oraclevcn.com). For more information, see DNS in Your Virtual Cloud Network.

f. **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

5. Click **Create Virtual Cloud Network**.
   The VCN is then created and displayed on the **Virtual Cloud Networks** page in the compartment you chose.

Next you'll typically want to create one or more subnets in the cloud network.

To create a subnet

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Click **Create Subnet**.
4. In the **Create Subnet** dialog box, you specify the resources to associate with the subnet (for example, a route table, and so on). By default, the subnet will be created in the current compartment, and you'll choose the resources from the same compartment.

Click the **click here** link in the dialog box if you want to enable compartment selection for the subnet and each of those resources.

Enter the following:

- **Create in Compartment:** If you've enabled compartment selection, specify the compartment where you want to put the subnet.
- **Name:** A friendly name for the subnet. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
- **Regional or AD-specific subnet:** Oracle recommends creating only regional subnets, which means that the subnet can contain resources in any of the region's availability domains. If you instead choose **Availability Domain-Specific** (the only kind of subnet that Oracle originally offered), you must also specify an availability domain. This choice means that any instances or other resources later created in this subnet must also be in that availability domain.
- **CIDR Block:** A single, contiguous CIDR block for the subnet (for example, 172.16.0.0/24). Make sure it's within the cloud network's CIDR block and doesn't overlap with any other subnets. You *cannot* change this value later. See Allowed VCN Size and Address Ranges. For reference, here's a CIDR calculator.
- **Route Table:** The route table to associate with the subnet. If you've enabled compartment selection, under **Route Table Compartment**, you must specify the compartment that contains the route table.
- **Private or public subnet:** This controls whether VNICs in the subnet can have public IP addresses. For more information, see Access to the Internet.
- **Use DNS Hostnames in this Subnet:** This option is available only if you provided a DNS label for the VCN during creation. If you want this subnet's instances to have DNS hostnames (which can be used with the built-in DNS capability in the VCN), select the check box for **Use DNS Hostnames in this Subnet**. Then you may specify a DNS label for the subnet, or the Console will generate one for you. The dialog box will automatically display the corresponding **DNS Domain Name** for the subnet (*<subnet DNS label>.<VCN DNS*

label>.oraclevcn.com). For more information, see DNS in Your Virtual Cloud Network.

- **DHCP Options:** The set of DHCP options to associate with the subnet. If you've enabled compartment selection, under **DHCP Options Compartment**, you must specify the compartment that contains the set of DHCP options.
- **Security Lists:** One or more security lists to associate with the subnet. If you've enabled compartment selection, you must specify the compartment that contains the security list.
- **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

5. Click **Create**.
   The subnet is then created and displayed on the **Subnets** page in the compartment you chose.

To edit a subnet

You can change these characteristics of a subnet:

- Name
- Which set of DHCP options the subnet uses
- Which route table the subnet uses
- Which security lists the subnet uses

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Click **Subnets**.

4. Click the subnet you're interested in.
5. Click **Edit**.
6. Make your changes.
7. Click **Save Changes**.
   The changes take effect within a few seconds.

To delete a subnet

Prerequisite: The subnet must have no instances, load balancers, DB systems, and orphaned mount targets in it. For more information, see Subnet or VCN Deletion.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Click **Subnets**.
4. Click the subnet you're interested in.
5. Click **Terminate**.
6. Confirm when prompted.

If the subnet is empty, its state changes to TERMINATING briefly and then TERMINATED. If the subnet is not empty, you get an error indicating that there are still instances or other resources in it that you must delete first.

To delete a VCN



Important

The Console has an easy "Delete all" process that

deletes a VCN and its related Networking resources (subnets, route tables, security lists, sets of DHCP options, internet gateway, and so on). If the VCN is attached to a dynamic routing gateway (DRG), the attachment is deleted, but the DRG remains.

The "Delete All" process deletes one resource at a time and takes a minute or two. A progress report is displayed to show you what's been deleted so far.

Before using the "Delete All" process, make sure there are no instances, load balancers, DB systems, or orphaned mount targets in any of the subnets. For more information, see Subnet or VCN Deletion.

If there are still resources in any subnet, or if you don't have permission to delete a particular Networking resource, the "Delete All" process stops and an error message is displayed. Any resources deleted up to that point cannot be restored. You might need to contact your tenancy administrator to help you delete any remaining resources.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Click **Terminate**.
   The resulting dialog box displays a list of the resources to be deleted. The list doesn't include the default components that come with the VCN, but they will be deleted along with the VCN.
4. Click **Delete All**.

The process begins. The progress is displayed as each resource is deleted.

5. When the process is complete, click **Close**.

To manage tags for a VCN

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Click the **Tags** tab to view or edit the existing tags. Or click **Apply tag(s)** to add new ones.

For more information, see Resource Tags.

To manage tags for a subnet

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Click the subnet you're interested in.
4. Click the **Tags** tab to view or edit the existing tags. Or click **Apply tag(s)** to add new ones.

For more information, see Resource Tags.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

To manage your VCNs, use these operations:

- ListVcns
- CreateVcn
- GetVcn
- UpdateVcn
- DeleteVcn: Deletes only the VCN and not its related resources. For more information, see Subnet or VCN Deletion. Note that the Console offers a "Delete All" process that makes it easy to delete the VCN and its related resources. See To delete a VCN.

To manage a VCN's subnets, use these operations:

- ListSubnets
- CreateSubnet
- GetSubnet
- UpdateSubnet
- DeleteSubnet

Ways to Secure Your Network

There are several ways you can control security for your cloud network and compute instances:

- **Public vs. private subnets:** You can designate a subnet to be private, which means instances in the subnet cannot have public IP addresses. For more information, see Public vs. Private Subnets.
- **Security lists:** To control packet-level traffic in/out of an instance. You configure security lists in the Oracle Cloud Infrastructure API or Console. For more information about security lists, see Security Lists.
- **Firewall rules:** To control packet-level traffic in/out of an instance. You configure firewall rules directly on the instance itself. Notice that Oracle Cloud Infrastructure images that run Oracle Linux automatically include default rules that allow ingress on

TCP port 22 for SSH traffic. Also, the Windows images include default rules that allow ingress on TCP port 3389 for Remote Desktop access. For more information, see Oracle-Provided Images.



Important

Firewall rules and security lists both operate at the instance level. However, you configure security lists at the subnet level, which means all instances in a given subnet have the same set of security list rules. Keep this in mind when setting up security for your cloud network and instances. When troubleshooting access to an instance, make sure both the security lists associated with the instance's subnet and the instance's firewall rules are set correctly.

If your instance is running Oracle Linux 7, you need to use firewalld to interact with the iptables rules. For your reference, here are commands for opening a port (1521 in this example):

```
sudo firewall-cmd --zone=public --permanent --add-
port=1521/tcp

sudo firewall-cmd --reload
```

- **Gateways and route tables:** To control general traffic flow from your cloud network to outside destinations (the internet, your on-premises network, or another VCN). You configure your cloud network's gateways and route tables in the Oracle Cloud Infrastructure API or Console. For more information about the gateways, see Networking Components. For more information about route tables, see Route Tables.
- **IAM policies:** To control who has access to the Oracle Cloud Infrastructure API or Console itself. You can control the type of access, and which cloud resources can be

accessed. For example, you can control who can set up your network and subnets, or who can update route tables or security lists. You configure policies in the Oracle Cloud Infrastructure API or Console. For more information, see Access Control.

Access Control

This topic gives basic information about using compartments and IAM policies to control access to your cloud network.

Compartments and Your Cloud Network

Anytime you create a cloud resource such as a virtual cloud network (VCN) or compute instance, you must specify which IAM *compartment* you want the resource in. A compartment is a collection of related resources that can be accessed only by certain groups that have been given permission by an administrator in your organization. The administrator will create compartments and corresponding IAM policies to control which users in your organization have access to which compartments. Ultimately, the goal is to ensure that each person has access to only the resources they need.

If your company is starting to try out Oracle Cloud Infrastructure, only a few people need to create and manage the VCN and its components, launch instances into the VCN, and attach block storage volumes to those instances. Those few people need access to *all* these resources, so all those resources could be in the same compartment.

In an enterprise production environment with a VCN, your company will want to use multiple compartments to make it easier to control access to certain types of resources. For example, your administrator could create Compartment_A for your VCN and other networking components. The administrator could then create Compartment_B for all the compute instances and block storage volumes that the HR organization uses, and Compartment_C for all the instances and block storage volumes that the Marketing organization uses. The administrator would then create IAM policies that give users only the level of access they need in each compartment. For example, the HR instance administrator is not entitled to modify the existing cloud network. So they would have full permissions for Compartment_B, but limited access to Compartment_A (just what's required to launch instances into the

network). If they tried to modify other resources in Compartment_A, the request would be denied.

For more information about compartments and how to control access to your cloud resources, see "Setting Up Your Tenancy" in the *Oracle Cloud Infrastructure Getting Started Guide* and Overview of Oracle Cloud Infrastructure Identity and Access Management.

IAM Policies for Networking

The simplest approach to granting access to Networking is the policy listed in Let network admins manage a cloud network. It covers the cloud network and all the other Networking components (subnets, security lists, route tables, gateways, and so on). To also give network admins the ability to launch instances (to test network connectivity), see Let users launch Compute instances.

If you're new to policies, see Getting Started with Policies and Common Policies.

For reference material for writing more detailed policies for Networking, see Details for the Core Services.

Individual Resource-Types

If you'd like, you can write policies that focus on individual resource-types (for example, just security lists) instead of the broader `virtual-network-family`. Be aware that the `instance-family` resource-type also includes several permissions for VNICs, which reside in a subnet but attach to an instance. For more information, see For instance-family Resource Types and Virtual Network Interface Cards (VNICs).

There's a resource-type called `local-peering-gateways` that is included within `virtual-network-family` and includes two other resource-types related to local VCN peering (within region):

- `local-peering-from`
- `local-peering-to`

The `local-peering-gateways` resource-type covers all permissions related to local peering gateways (LPGs). The `local-peering-from` and `local-peering-to` resource-types are for

granting permission to *connect* two LPGs and form a peering relationship within a single region. For more information, see Local VCN Peering (Within Region).

Similarly, there's a resource-type called `remote-peering-connections` that is included within `virtual-network-family` and includes two other resource-types related to remote VCN peering (across regions):

- `remote-peering-from`
- `remote-peering-to`

The `remote-peering-connections` resource-type covers all permissions related to remote peering connections (RPCs). The `remote-peering-from` and `remote-peering-to` resource-types are for granting permission to *connect* two RPCs and form a peering relationship across regions. For more information, see Remote VCN Peering (Across Regions).

Nuances of Different Verbs

If you'd like, you can write policies that limit the level of access by using a different policy verb (`manage` versus `use`, and so on). If you do, there are some nuances to understand about the policy verbs for Networking.

Be aware that the `inspect` verb not only returns general information about the cloud network's components (for example, the name and OCID of a security list, or of a route table). It also includes the contents of the component (for example, the actual rules in the security list, the routes in the route table, and so on).

Also, the following types of abilities are available only with the `manage` verb, not the `use` verb:

- Update (enable/disable) `internet-gateways`
- Update `security-lists`
- Update `route-tables`
- Update `dhcp-options`
- Attach a dynamic routing gateway (DRG) to a virtual cloud network (VCN)
- Create an IPSec connection between a DRG and customer-premises equipment (CPE)
- Peer VCNs



Important

Each VCN has various components that directly affect the behavior of the network (route tables, security lists, DHCP options, Internet Gateway, and so on). When you create one of these components, you establish a relationship between that component and the VCN, which means you must be allowed in a policy to both create the component and manage the VCN itself. However, the ability to *update* that component (to change the route rules, security list rules, and so on) does NOT require permission to manage the VCN itself, even though changing that component can directly affect the behavior of the network. This discrepancy is designed to give you flexibility in granting least privilege to users, and not require you to grant excessive access to the VCN just so the user can manage other components of the network. Be aware that by giving someone the ability to update a particular type of component, you're implicitly trusting them with controlling the network's behavior.

For more information about policy verbs, see Verbs.

Security Lists

In addition to using IAM policies to control who can manipulate your VCN (for example, add an internet gateway, change route table rules), you can use *security lists* to control traffic at the packet level.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Overview of Security Lists

A security list provides a virtual firewall for an instance, with ingress and egress rules that specify the types of traffic allowed in and out. Each security list is enforced at the instance level. However, you configure your security lists *at the subnet level*, which means that all instances in a given subnet are subject to the same set of rules. The security lists apply to a given instance whether it's talking with another instance in the VCN or a host outside the VCN.

Each subnet can have multiple security lists associated with it, and each list can have multiple rules (for the maximum number, see Service Limits). A packet in question is allowed if *any rule in any of the lists* allows the traffic (or if the traffic is part of an existing connection being tracked). There's a caveat if the lists happen to contain both stateful and stateless rules that cover the same traffic. For more information, see Stateful vs. Stateless Rules.



Important

Your instances running Oracle-provided Linux images or Windows images also have firewall rules that control access to the instance. When troubleshooting access to an instance, make sure both the security lists associated with the instance's subnet and the instance's firewall rules are set correctly. For more information, see Oracle-Provided Images.

If your instance is running Oracle Linux 7, you need to use firewalld to interact with the iptables rules. For your reference, here are commands for opening a port (1521 in this example):

```
sudo firewall-cmd --zone=public --permanent --add-
port=1521/tcp

sudo firewall-cmd --reload
```

Security lists are regional entities. For the limit on the number of security lists you can have in a VCN, see Service Limits.



Note

Security lists are not enforced for traffic involving the 169.254.0.0/16 CIDR block, which includes services such as iSCSI and instance metadata.

Stateful vs. Stateless Rules

When you create a security list rule, you choose whether it's *stateful* or *stateless*. The difference is described below. The default is stateful. Stateless rules are recommended if you

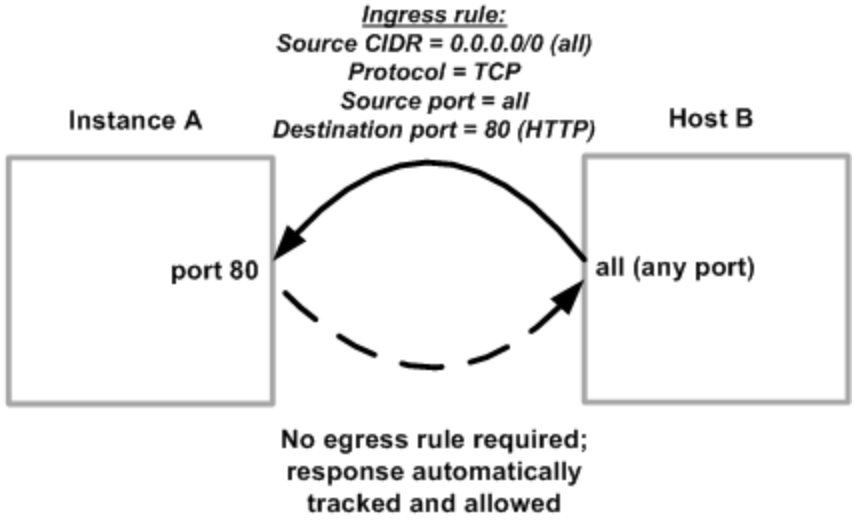
have a high-volume internet-facing website (for the HTTP/HTTPS traffic).

Stateful Rules

Marking a security list rule as stateful indicates that you want to use connection tracking for any traffic that matches that rule (for instances in the subnet the security list is associated with). This means that when an instance receives traffic matching the stateful ingress rule, the response is tracked and automatically allowed back to the originating host, regardless of any egress rules applicable to the instance. And when an instance sends traffic that matches a stateful egress rule, the incoming response is automatically allowed, regardless of any ingress rules. For more details, see Connection Tracking Details for Stateful Rules.

The figure below illustrates a stateful ingress rule for an instance that needs to receive and respond to HTTP traffic. Instance A and Host B are communicating (Host B could be any host, whether an instance or not). The stateful ingress rule allows traffic from any source IP address (0.0.0.0/0) to destination port 80 only (TCP protocol). No egress rule is required to allow the response traffic.

Stateful Security List: Receive HTTP Traffic

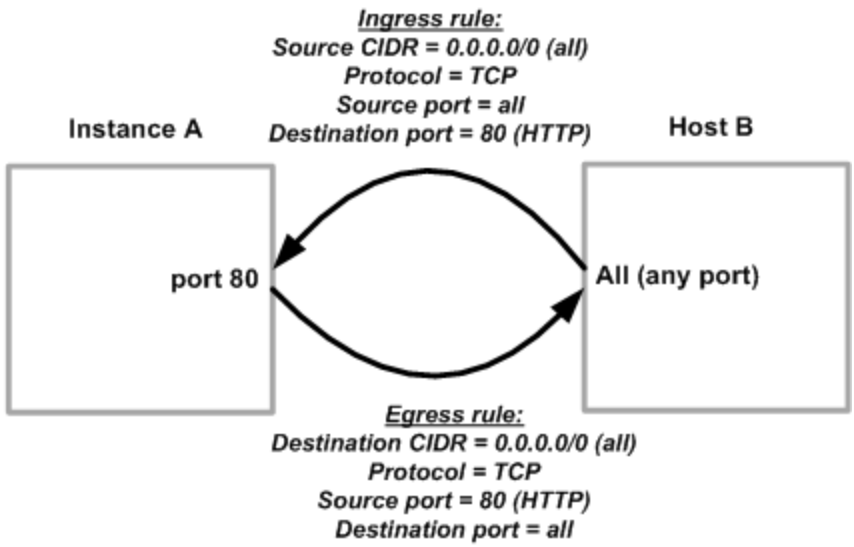


Stateless Rules

Marking a security list rule as stateless indicates that you do NOT want to use connection tracking for any traffic that matches that rule (for instances in the subnet the security list is associated with). This means that response traffic is not automatically allowed. To allow the response traffic for a stateless ingress rule, you must create a corresponding stateless egress rule.

The next figure shows Instance A and Host B as before, but now with stateless security list rules. As with the stateful rule above, the stateless ingress rule allows traffic from all IP addresses and any ports, on destination port 80 only (using the TCP protocol). To allow the response traffic, there needs to be a corresponding stateless egress rule that allows traffic to any destination IP address (0.0.0.0/0) and any ports, from source port 80 only (using the TCP protocol).

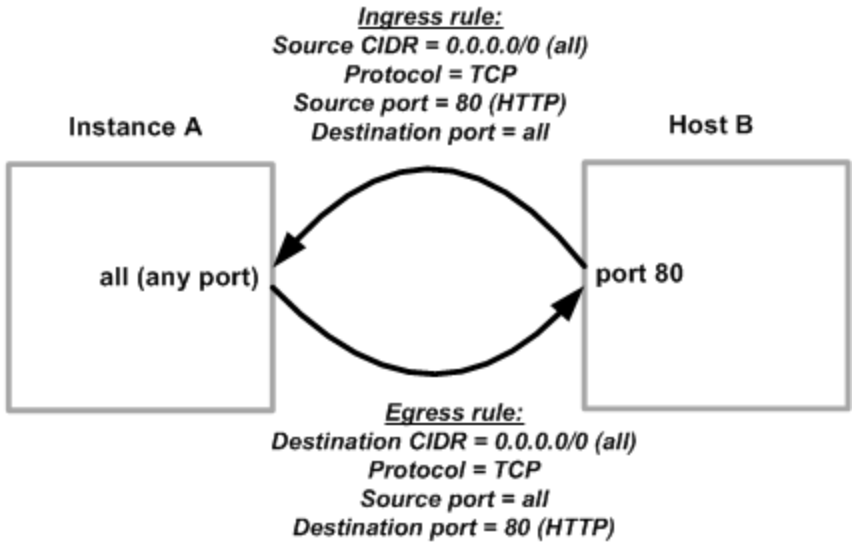
Stateless Security List: Receive HTTP Traffic



If Instance A needs instead to *initiate* HTTP traffic and get the response, then a different set of stateless rules are required. As the next figure shows, the egress rule would have source port

= all and destination port = 80 (HTTP). The ingress rule would then have source port 80 and destination port = all.

Stateless Security List: Initiate HTTP Traffic



If you were to use port binding on Instance A to specify exactly which port the HTTP traffic would come from, you could specify that as the source port in the egress rule and the destination port in the ingress rule.



Note

If for some reason your security lists contain both stateful and stateless rules, and there's traffic that matches both a stateful and stateless rule in a particular direction (for example, ingress), the stateless rule takes precedence and the connection is not tracked. You would need a corresponding rule in the other direction (for example, egress, either stateless or stateful) in order for the response traffic to be allowed.

Enabling Path MTU Discovery Messages for Stateless Rules

If you decide to use stateless security list rules to allow traffic to/from endpoints outside the VCN, it's important to add a security list rule that allows ingress ICMP traffic type 3 code 4 from source 0.0.0.0/0 and any source port. This rule enables your instances to receive Path MTU Discovery fragmentation messages. This rule is critical for establishing a connection to your instances. Without it, you can experience connectivity issues. For more information, see Hanging Connection.

Default Security List

Each cloud network has a *default security list*. A given subnet automatically has the default security list associated with it if you don't specify one or more other security lists during subnet creation. At any time after you create a subnet, you can change which security lists are associated with it. And you can change the rules in the lists.

Unlike other security lists, the default security list comes with an initial set of stateful rules, which you can change:

- **Stateful ingress:** Allow TCP traffic on destination port 22 (SSH) from source 0.0.0.0/0 and any source port. This rule makes it easy for you to create a new cloud network and

public subnet, launch a Linux instance, and then immediately connect via SSH to that instance without needing to write any security list rules yourself.



Important

The default security list does not include a rule to allow Remote Desktop Protocol (RDP) access. If you're using Windows images, make sure to add a stateful ingress rule for TCP traffic on destination port 3389 from source 0.0.0.0/0 and any source port.
See To enable RDP access for more information.

- **Stateful ingress:** Allow ICMP traffic type 3 code 4 from source 0.0.0.0/0. This rule enables your instances to receive Path MTU Discovery fragmentation messages.
- **Stateful ingress:** Allow ICMP traffic type 3 (all codes) from source = your VCN's CIDR. This rule makes it easy for your instances to receive connectivity error messages from other instances within the VCN.
- **Stateful egress:** Allow all traffic. This allows instances to initiate traffic of any kind to any destination. Notice that this means the instances with public IP addresses can talk to any internet IP address if the VCN has a configured internet gateway. And because stateful security rules use connection tracking, the response traffic is automatically allowed regardless of any ingress rules. For more information, see Connection Tracking Details for Stateful Rules.

The default security list comes with no stateless rules. However, you can add or remove rules from the default security list as you like.

Enabling Ping

The default security list does not include a rule to allow ping requests. If you want to ping an instance, make sure the instance's security lists include an *additional* stateful ingress rule to specifically allow ICMP traffic type 8 from the source network you plan to ping from. To allow

ping access from the internet, use 0.0.0.0/0 for the source. Note that this rule for pinging is separate from the default ICMP-related rules in the default security list. Don't remove those rules.

Parts of a Security List Rule

Each security list can contain one or more rules, and each rule **allows** either ingress or egress traffic. You choose whether the rule is stateful or stateless. For examples of rules, see Stateful vs. Stateless Rules, and see Networking Scenarios. For the limit on the number of rules you can have in a security list, see Service Limits.

Each ingress rule specifies the following:

- **Stateful or stateless:** If stateful, connection tracking is used for traffic matching the rule. If stateless, no connection tracking is used. See Stateful vs. Stateless Rules.
- **Source Type:** Possible values:
  - **CIDR:** The typical choice.
  - **Service CIDR:** Only for traffic coming from an Oracle service through a service gateway.
- **Source CIDR:** Only if the **Source Type** is **CIDR** (the typical choice). The source CIDR is the CIDR block where the traffic originates from. Use 0.0.0.0/0 to indicate all IP addresses. The prefix is required (for example, include the /32 if specifying an individual IP address).
- **Source Service:** Only if the **Source Type** is **Service CIDR** (which means the traffic is coming from an Oracle service through a service gateway). The source service is the service CIDR label that you're interested in.
- **IP Protocol:** Either a single IPv4 protocol or "all" to cover all protocols.
- **Source port:** The port where the traffic originates from. For TCP or UDP, you can specify all source ports, or optionally specify a single source port number, or a range.
- **Destination port:** The port where the traffic is destined to. For TCP or UDP, you can specify all destination ports, or optionally specify a single destination port number, or a range.

- **ICMP type and code:** For ICMP, you can specify all types and codes, or optionally specify a single type with an optional code. If the type has multiple codes, create a separate rule for each code you want to allow.

Each egress rule specifies the following:

- **Stateful or stateless:** Whether connection tracking is used for the traffic matching the rule. See Stateful vs. Stateless Rules.
- **Destination Type:** Possible values:
  - **CIDR:** The typical choice.
  - **Service CIDR:** Only for traffic going to an Oracle service through a service gateway.
- **Destination CIDR:** Only if the **Destination Type** is **CIDR** (the typical choice).The destination CIDR is the CIDR block where the traffic is destined to. Use 0.0.0.0/0 to indicate all IP addresses. The prefix is required (for example, include the /32 if specifying an individual IP address).
- **Destination Service:** Only if the **DestinationType** is **Service CIDR** (which means the traffic is going to an Oracle service through a service gateway). The destination service is the service CIDR label that you're interested in.
- **IP Protocol:** Either a single IPv4 protocol or "all" to cover all protocols.
- **Source port:** The port where the traffic originates from. For TCP or UDP, you can specify all source ports, or optionally specify a single source port number, or a range.
- **Destination port:** The port where the traffic is destined to. For TCP or UDP, you can specify all destination ports, or optionally specify a single destination port number, or a range.
- **ICMP type and code:** For ICMP, you can specify all types and codes, or optionally specify a single type with an optional code. If the type has multiple codes, create a separate rule for each code you want to allow.

For instructions on working with security lists and rules, see the sections that follow.

Connection Tracking Details for Stateful Rules

Oracle uses connection tracking to allow responses for traffic that matches stateful rules (see Stateful vs. Stateless Rules). Each instance has a maximum number of concurrent connections that can be tracked, based on the instance's shape.

To determine response traffic for TCP, UDP, and ICMP, Oracle performs connection tracking on these items for the packet:

- Protocol
- Source and destination IP addresses
- Source and destination ports (for TCP and UDP only)



Note

For other protocols, Oracle tracks only the protocol and IP addresses, and not the ports. This means that when an instance initiates traffic to another host and that traffic is allowed by egress security list rules, any traffic that the instance subsequently receives from that host for a period is considered response traffic and is allowed.

Rules to Handle Fragmented UDP Packets

Instances can send or receive UDP traffic. If a UDP packet is too large for the connection, it is fragmented. However, only the first fragment from the packet contains the protocol and port information. If the security list rules that allow this ingress or egress traffic specify a particular port number (source or destination), the fragments after the first one are dropped. If you expect your instances to send or receive large UDP packets, set both the source and destination ports for the applicable security list rules to ALL (instead of a particular port number).

Working with Security Lists

When you create a new subnet, you must associate at least one security list with it. It can be either the VCN's default security list or one or more other security lists that you've created (for the maximum number, see Service Limits). You can change which security lists the subnet uses at any time.

You may optionally assign a friendly name to the security list during creation. It doesn't have to be unique, and you can change it later. Oracle automatically assigns the security list a unique identifier called an Oracle Cloud ID (OCID). For more information, see Resource Identifiers.

For the purposes of access control, you must specify the compartment where you want the security list to reside. Consult an administrator in your organization if you're not sure which compartment to use. For more information, see Access Control.

You can add and remove rules from the security list, but notice that when you update a security list in the API, the new set of rules replaces the entire existing set of rules.

To delete a security list, it must not be associated with a subnet yet. You can't delete a VCN's default security list.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let network admins manage a cloud network covers management of all Networking components, including security lists.

If you have security admins who need to manage security lists but not other components in Networking, you could write a more restrictive policy:

```
Allow group SecListAdmins to manage security-lists in tenancy

Allow group SecListAdmins to manage vcns in tenancy
```

Both statements are needed because the creation of a security list affects the VCN the security list is in. The second statement also allows the SecListAdmins group to create new VCNs, but not create subnets or manage any other components related to any of those VCNs, except for the security lists. The group also can't delete any existing VCNs that already have subnets in them.

For more information, see IAM Policies for Networking.

Using the Console

To view a VCN's default security list

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Under **Resources**, click **Security Lists.**
4. Click the security list to view its details.
   Under **Resources**, you can click **Ingress Rules** or **Egress Rules** to switch between the different types of rules.

To update rules in an existing security list



Important

When updating the default security list, be aware of the default rules, the purpose each serves, and the consequences of deleting them. For example, deleting the existing ICMP type 3 code 4 rule can cause problems when you connect to your instances. For more information, see Hanging Connection.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Under **Resources**, click **Security Lists**.
4. Click the security list you're interested in.
5. Under **Resources**, click either **Ingress Rules** or **Egress Rules** depending on the type of rule you want to work with.
6. If you want to add a new rule, click **Add Ingress Rule** (or **Add Egress Rule**). See details of adding a rule in To create a new security list.
7. If you want to delete an existing rule, click the Actions icon (three dots), and then click **Remove**.
8. If you wanted to edit an existing rule, click the Actions icon (three dots), and then click **Edit**.

To create a new security list

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Under **Resources**, click **Security Lists**.
4. Click **Create Security List**.
5. Enter the following:
    a. **Name:** A friendly name for the security list. The name doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
    b. **Create in Compartment:** The compartment where you want to create the security list, if different from the compartment you're currently working in.
6. Add either an ingress rule or egress rule (for examples of rules, see Networking

Scenarios):

a. Click either **Add Ingress Rule** or **Add Egress Rule**.

b. Choose whether it's a stateful or stateless rule (see Stateful vs. Stateless Rules). By default, rules are stateful unless you specify otherwise.

c. Enter either the source CIDR (for ingress) or destination CIDR (for egress). For example, use 0.0.0.0/0 to indicate all IP addresses. Other typical CIDRs you might specify in a rule are the CIDR block for your on-premises network, or for a particular subnet. If you're setting up a security list rule to allow traffic with a service gateway, instead see Task 3: (Optional) Update the security lists for the subnet.

d. Select the IP protocol (for example, TCP, UDP, ICMP, "All protocols", and so on).

e. Enter further details depending on the protocol:

- If you chose TCP or UDP, enter a source port range and destination port range. You can enter "All" to cover all ports. If you want to allow a specific port, enter the port number (for example, 22 for SSH or 3389 for RDP) or a port range (for example, 20-22).
- If you chose ICMP, you can enter "All" to cover all types and codes. If you want to allow a specific ICMP type, enter the type and an optional code separated by a comma (for example, 3,4). If the type has multiple codes you want to allow, create a separate rule for each code.

7. Repeat the preceding step for each rule you want to add to the list.
8. **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
9. When you're done, click **Create Security List**.

The security list is created and then displayed on the **Security Lists** page in the compartment you chose. You can now specify this security list when creating or updating a subnet.

When you view all the rules in a security list, notice that any stateless rules in the list are shown above any stateful rules. Stateless rules in the list take precedence over stateful rules. In other words: If there's traffic that matches both a stateless rule and a stateful rule across all the security lists associated with the subnet, the stateless rule takes precedence and the connection is not tracked.

To change which security lists a subnet uses

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Click **Subnets**.
4. Click the subnet you're interested in.
5. Under **Resources**, click **Security Lists**.
6. If you want to add a security list, click **Add Security List**, and select the new security list you want the subnet to use.
7. If you want to remove a security list, click the Actions icon (three dots), and then click **Remove**. Remember that a subnet must always have at least one security list associated with it.

   The changes take effect within a few seconds.

To delete a security list

Prerequisite: To delete a security list, it must not be associated with a subnet yet. You can't delete the default security list in a VCN.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Under **Resources**, click **Security Lists**.

4. For the security list you want to delete, click the Actions icon (three dots), and then click **Terminate**.
5. Confirm when prompted.

To manage tags for a security list

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Under **Resources**, click **Security Lists**.
4. Click the security list you're interested in.
5. Click the **Tags** tab to view or edit the existing tags. Or click **Apply tag(s)** to add new ones.

For more information, see Resource Tags.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

To manage a VCN's security lists, use these operations:

- ListSecurityLists
- GetSecurityList
- UpdateSecurityList
- CreateSecurityList
- DeleteSecurityList

Virtual Network Interface Cards (VNICs)

This topic describes how to manage the virtual network interface cards (VNICs) in a virtual cloud network (VCN).



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Overview of VNICs and Physical NICs

The servers in Oracle Cloud Infrastructure data centers have physical network interface cards (NICs). When you launch an instance on one of these servers, the instance communicates using Networking service *virtual* NICs (VNICs) associated with the physical NICs. The next sections talk about VNICs and NICs, and how they're related.

About VNICs

A VNIC enables an instance to connect to a VCN and determines how the instance connects with endpoints inside and outside the VCN. Each VNIC resides in a subnet in a VCN and includes these items:

- One *primary* private IPv4 address from the subnet the VNIC is in, chosen by either you or Oracle.
- Up to 31 optional *secondary* private IPv4 addresses from the same subnet the VNIC is in, chosen by either you or Oracle.
- An optional public IPv4 address for each private IP, chosen by Oracle but assigned by you at your discretion.

- An optional hostname for DNS for each private IP address (see DNS in Your Virtual Cloud Network).
- A MAC address.
- A VLAN tag assigned by Oracle and available when attachment of the VNIC to the instance is complete (relevant only for bare metal instances).
- A flag to enable or disable the source/destination check on the VNIC's network traffic (see Source/Destination Check).

Each VNIC also has a friendly name you can assign, and an Oracle-assigned OCID (see Resource Identifiers).

Each instance has a *primary VNIC* that is automatically created and attached during launch. That VNIC resides in the subnet you specify during launch. The primary VNIC cannot be removed from the instance.

How VNICs and Physical NICs Are Related

This section is relevant to bare metal instances.

The OS on a bare metal instance recognizes two physical network devices and configures them as two physical NICs, 0 and 1. Whether they're both active depends on the underlying hardware:

- **Oracle X5 servers (also called *first-generation*):** Only NIC 0 is active.
- **Oracle X7 servers (also called *second-generation*):** Both NIC 0 and NIC 1 are active. Each physical NIC has 25 Gbps bandwidth.

NIC 0 is automatically configured with the primary VNIC's IP configuration (the IP addresses, DNS hostname, and so on).

If you add a *secondary VNIC* to a second-generation instance, you must specify which physical NIC the secondary VNIC should use. You must also configure the OS so that the physical NIC has the secondary VNIC's IP configuration. For Linux instances, see Linux: Configuring the OS for Secondary VNICs. For Windows instances, see Windows: Configuring the OS for Secondary VNICs.

About Secondary VNICs

You can add secondary VNICs to an instance after it's launched. Each secondary VNIC can be in a subnet in the same VCN as the primary VNIC, or in a different subnet that is either in the same VCN or a different one. However, all the VNICs must be in the same availability domain as the instance.

Here are some reasons why you might use secondary VNICs:

- **Use your own hypervisor on a bare metal instance:** The virtual machines on the bare metal instance each have their own secondary VNIC, giving direct connectivity to other instances and services in the VNIC's VCN. For more information, see Installing and Configuring KVM on Bare Metal Instances with Multi-VNIC.
- **Connect an instance to subnets in multiple VCNs:** For example, you might set up your own firewall to protect traffic between VCNs, so the instance needs to connect to subnets in different VCNs.

Here are more details about secondary VNICs:

- They're supported for these types of instances:
  - **Linux:** Both VM and bare metal instances. Also see Linux: Configuring the OS for Secondary VNICs.
  - **Windows:** Both VM and bare metal instances, but only on X7/second-generation shapes (shapes with "2" in the name, such as VM.Standard 2.16 and BM.Standard2.52). For bare metal, secondary VNICs are supported only on the second physical NIC. Remember that the first physical NIC is NIC 0, and the second is NIC 1. Also see Windows: Configuring the OS for Secondary VNICs.
- There's a limit to how many VNICs can be attached to an instance, and it varies by shape. For those limits, see Compute Shapes.
- They can be added only after the instance is launched.
- They must always be attached to an instance and cannot be moved. The process of creating a secondary VNIC automatically attaches it to the instance. The process of detaching a secondary VNIC automatically deletes it.
- They are automatically detached and deleted when you terminate the instance.

- The instance's bandwidth is fixed regardless of the number of VNICs attached. You can't specify a bandwidth limit for a particular VNIC on an instance.
- Attaching multiple VNICs from the same subnet CIDR block to an instance can introduce asymmetric routing, especially on instances using a variant of Linux. If you need this type of configuration, Oracle recommends assigning multiple private IP addresses to one VNIC, or using policy-based routing as shown in the script later in this topic.

Source/Destination Check

By default, every VNIC performs the source/destination check on its network traffic. The VNIC looks at the source and destination listed in the header of each network packet. If the VNIC is not the source or destination, then the packet is dropped.

If the VNIC needs to forward traffic (for example, if it needs to perform Network Address Translation (NAT)), you must disable the source/destination check on the VNIC. For instructions, see To update an existing VNIC. For information about the general scenario, see Using a Private IP as a Route Target.

VNIC Information in the Instance Metadata

The instance metadata includes information about the VNICs at this URL:

```
http://169.254.169.254/opc/v1/vnics/
```

Here's an example response:

```
[ {
  "vnicId" : "ocid1.vnic.oc1.phx.examplevq7kncmdtfr23dznohdkd2cywjcem33eg3dxa",
  "privateIp" : "10.0.3.6",
  "vlanTag" : 11,
  "macAddr" : "02:00:17:00:12:D3",
  "virtualRouterIp" : "10.0.3.1",
  "subnetCidrBlock" : "10.0.3.0/24",
  "nicIndex" : 0
}, {
  "vnicId" : "ocid1.vnic.oc1.phx.exampledfslcmdyepqc73ntv25ft3rqxsooplb4l2zg7q",
  "privateIp" : "10.0.4.3",
  "vlanTag" : 12,
  "macAddr" : "02:00:17:00:13:13",
```

```
  "virtualRouterIp" : "10.0.4.1",
  "subnetCidrBlock" : "10.0.4.0/24",
  "nicIndex" : 0
} ]
```

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: see IAM Policies for Networking.

Monitoring Resources

You can monitor the health, capacity, and performance of your Oracle Cloud Infrastructure resources by using metrics, alarms, and notifications. For more information, see Monitoring Overview and Notifications Overview.

For information about monitoring the traffic coming in and out of VNICs, see Networking Metrics.

Using the Console

To view an instance's VNICs

1. Confirm you're viewing the compartment that contains the instance you're interested in.
2. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
3. Click the instance to view its details.

4. Under **Resources**, click **Attached VNICs**.
   The primary VNIC and any secondary VNICs attached to the instance are displayed. If the instance has two active physical NICs, the VNICs are grouped by NIC 0 and NIC 1.

To create and attach a secondary VNIC

1. Confirm you're viewing the compartment that contains the instance you're interested in.
2. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
3. Click the instance to view its details.
4. Under **Resources**, click **Attached VNICs**.
   The primary VNIC and any secondary VNICs attached to the instance are displayed.
5. Click **Create VNIC**.
6. In the **Create VNIC** dialog box, you specify which VCN and subnet to put the VNIC in. By default, the VNIC will be created in the current compartment, and you'll choose the VCN and subnet from the same compartment. Click the **click here** link in the dialog box if you want to enable compartment selection and choose a VCN or subnet in a different compartment.
   Enter the following:
   - **Name:** A friendly name for the secondary VNIC. The name doesn't have to be unique, and you can change it later. Avoid entering confidential information.
   - **Virtual Cloud Network Compartment:** The compartment that contains the VCN that in turn contains the subnet of interest.
   - **Virtual Cloud Network:** The VCN that contains the subnet of interest.
   - **Subnet Compartment:** The compartment that contains the subnet of interest.
   - **Subnet:** The subnet of interest. The secondary VNIC must be in the same availability domain as the instance's primary VNIC, so the subnet list includes any regional subnets or AD-specific subnets in the primary VNIC's availability domain.

- **Physical NIC:** Only relevant if this is a bare metal instance with two active physical NICs. Select which one you want the secondary VNIC to use. When you later view the instance's details and the list of VNICs attached to the instance, they'll be grouped by NIC 0 and NIC 1.
- **Skip Source/Destination Check:** By default, this check box is NOT selected, which means the VNIC performs the source/destination check. Only select this check box if you want the VNIC to be able to forward traffic. See Source/Destination Check.
- **Private IP Address:** Optional. An available private IP address of your choice from the subnet's CIDR (otherwise the private IP address is automatically assigned).
- **Assign public IP address:** Whether to assign an ephemeral public IP address to the VNIC's primary private IP. Available only if the subnet is public. For more information, see Public IP Addresses.
- **Hostname:** Optional. A hostname to be used for DNS within the cloud network. Available only if the VCN and subnet both have DNS labels. For more information, see DNS in Your Virtual Cloud Network.
- **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

7. Click **Create VNIC**.
   The secondary VNIC is created and then displayed on the **Attached VNICs** page for the instance. It can take several seconds before the secondary VNIC appears on the page.
8. Configure the OS to use the VNIC. See Linux: Configuring the OS for Secondary VNICs or Windows: Configuring the OS for Secondary VNICs.

To update an existing VNIC

You can update the VNIC's friendly name or hostname, or whether the VNIC performs the source/destination check.

1. Confirm you're viewing the compartment that contains the instance you're interested in.
2. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
3. Click the instance to view its details.
4. Under **Resources**, click **Attached VNICs**.
   The primary VNIC and any secondary VNICs attached to the instance are displayed.
5. For the VNIC you want to edit, click the Actions icon (three dots), and then click **Edit VNIC**.
6. Make your changes and click **Update VNIC**.

To delete a secondary VNIC



Warning

If the VNIC has a private IP that is the target of a route rule, deleting the VNIC causes the route rule to blackhole and traffic will be dropped.

1. Confirm you're viewing the compartment that contains the instance you're interested in.
2. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
3. Click the instance to view its details.
4. Under **Resources**, click **Attached VNICs**.
   The primary VNIC and any secondary VNICs attached to the instance are displayed.

5. For the VNIC you want to delete, click the Actions icon (three dots), and then click **Delete**.
6. Confirm when prompted.

It takes typically a few seconds before the VNIC is deleted.

If the secondary VNIC is on a Linux instance: If you then run the provided script in Linux: Configuring the OS for Secondary VNICs, it removes the secondary VNIC from the OS configuration.

To manage tags for a VNIC

1. Confirm you're viewing the compartment that contains the instance you're interested in.
2. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
3. Click the instance to view its details.
4. Under **Resources**, click **Attached VNICs**.
   The primary VNIC and any secondary VNICs attached to the instance are displayed.
5. Click the VNIC you're interested in.
6. Click the **Tags** tab to view or edit the existing tags. Or click **Apply tag(s)** to add new ones.

For more information, see Resource Tags.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

To manage VNICs on an instance, use these operations:

- ListVnicAttachments: Use this to list the VNICs attached to an instance.
- GetVnicAttachment: Use this to get the VNIC's VLAN tag and other properties.
- GetVnic: Use this to get the VNIC's private IP address, MAC address, optional public IP address, optional DNS hostname, and other properties.
- AttachVnic
- DetachVnic
- UpdateVnic

Linux: Configuring the OS for Secondary VNICs

This section gives details about OS configuration that is required for secondary VNICs on instances that run a variant of Linux.

At the end of the section is a script that you can use to configure secondary VNICs on either VM instances or bare metal instances.

Linux VM Instances

When you add a secondary VNIC to a Linux VM instance, a new interface (that is, an Ethernet device) is added to the instance and automatically recognized by the OS. However, DHCP is not active for the secondary VNIC, and you must configure the interface with the static IP address and default route. The script provided here handles that configuration for you.

Linux Bare Metal Instances

When you add a secondary VNIC to a Linux bare metal instance, the OS does not automatically recognize the secondary VNIC, so you must configure it in the OS. Depending on your requirements, you can configure it as either:

- An SR-IOV virtual function (see Installing and Configuring KVM on Bare Metal Instances with Multi-VNIC).
- A VLAN-tagged interface (see the script in the following section).

Configuration Script for Either Linux VM Instances or Linux Bare Metal Instances

The following script works for both VM instances and bare metal instances. It looks at the secondary VNIC information in the instance metadata and configures the OS accordingly. You could run the script periodically to bring the OS configuration up to date with the instance metadata.

For VM instances in particular, the OS automatically recognizes the secondary VNIC's interface, and the script just needs to configure the static IP address and default route.

For bare metal instances in particular, the script creates the interface for the secondary VNIC and configures it with the relevant information. If the instance has two active physical NICs (an X7/second-generation shape with NIC 0 and NIC 1), the script configures the secondary VNIC to use whichever physical NIC you chose when you added the VNIC to the instance. Note that for NIC 1, if a secondary VNIC has VLAN tag 0, it uses the NIC's interface. The script doesn't create an interface for that secondary VNIC.

Here are some additional notes about how the script works for both VM instances and bare metal instances:

- **Default namespace and policy-based routing:** By default, this script configures the secondary VNIC in the default namespace and with policy-based routing so applications can communicate through the VNIC with hosts outside the VNIC's subnet. **This policy-based routing has effect only if the packets are sourced from the IP address of the secondary VNIC.** The ability to bind to a specific source IP address or source interface exists in most tools (such as ssh, ping, and wget) and libraries that initiate connections. For example, the `ssh -b` option lets you bind to the private IP address of the secondary VNIC:

  ```
  ssh -b <secondary_VNIC_IP_address> <destination_IP_address>
  ```

  Be aware that if traffic comes in to a service on the instance through a secondary VNIC's interface and the service replies, the reply packets automatically have the VNIC's interface IP address as the source IP address. Policy-based routing is required for that reply to go back out on the same interface and find the correct default gateway.
- **A separate namespace:** If you're familiar with namespaces, you can instead configure the secondary VNIC in another namespace of your choice by running the script

with the `-n` option. A separate namespace is required when an instance has secondary VNICs that are attached to subnets in different VCNs, and those subnets have overlapping CIDR blocks.

- **Secondary private IPs:** The instance metadata does not include information about any secondary private IPs assigned to the instance. To include that as part of the script's OS configuration, you must provide the secondary private IP address and OCID at the command line when you run the script.
- **Removal of a secondary VNIC:** After deleting a secondary VNIC from an instance, running the script removes the VNIC's information from the OS configuration.



Important

The script uses a simple configuration process that does not persist if you reboot the instance. If you use the script, make sure to rerun it after each reboot.

Here are basic examples of how to run the script:

- *<script_name>* `-c`: Configure (adds or deletes) secondary VNIC host IP configuration
- *<script_name>* `-c -n`: Same but uses separate namespaces
- *<script_name>* `-d`: Force removes all secondary VNIC host IP configuration

See the script's help for more information.



Tip

Download the script from the online version of this user guide at https://docs.cloud.oracle.com/iaas/Content/Network/Tasks/managingVNICs.htm#linux.

Windows: Configuring the OS for Secondary VNICs

Secondary VNICs are supported on VM and bare metal instances, **but only on X7/second-generation shapes** (shapes with "2" in the name, such as VM.Standard2.16 and BM.Standard2.52). For bare metal, secondary VNICs are supported **only on the second physical NIC**.



Tip

The first physical NIC is NIC 0, and the second is NIC 1.

You must configure the secondary VNIC within the OS. There's an Oracle-provided PowerShell script that performs configuration. When running the script, you can optionally provide the secondary VNIC's OCID (which you can get from the instance's VNIC metadata):

```
.\secondary_vnic_windows_configure.ps1 "<secondary_VNIC_OCID>"
```

Otherwise, the script shows a list of the secondary VNICs on the instance and asks you to choose the one you want to configure. Here's generally what the script does:

1. The script checks if the network interface has an IP address and a default route.
2. To enable the OS to recognize the secondary VNIC, the **script must overwrite the IP address and default route with static settings (which effectively disables DHCP).** The script prompts you with a choice: to overwrite with the static settings, or exit.



Tip

Download the script from the online version of this user guide at https://docs.cloud.oracle.com/iaas/Content/Network/Tasks/managingVNICs.htm#windows.

The overall process for configuration varies slightly depending on the type of instance (VM or bare metal) and how many secondary VNICs you add to the instance.

Windows VM instances

Here's the overall process:

1. Add one or more secondary VNICs to the instance. Keep each VNIC's OCID handy so you can provide it later when you run the configuration script. You can also get the OCID from the instance's VNIC metadata.
2. Connect to the instance with Remote Desktop.
3. Run the script:
    a. Open PowerShell as an administrator.
    b. Run the script with the secondary VNIC's OCID:

    ```
    .\secondary_vnic_windows_configure.ps1 "<secondary_VNIC_OCID>"
    ```
4. Repeat the preceding step for each additional secondary VNIC.

Windows bare metal instances: adding the first secondary VNIC

If you're adding only a single secondary VNIC to the bare metal instance, here's the overall process:

1. Add the secondary VNIC to your instance. Keep the VNIC's OCID handy so you can provide it when later running the configuration script. You can also get the OCID from the instance's VNIC metadata.
2. Connect to the instance with Remote Desktop.
3. Enable the second physical NIC on the instance:
    a. Open the Device Manager, and then click **Network adapters**.
    b. Right-click the adapter that corresponds to the instance's second physical NIC, and click **Enable**.

4. Run the script:
    a. Open PowerShell as an administrator.
    b. Run the script with the secondary VNIC's OCID:

    ```
    .\secondary_vnic_windows_configure.ps1 "<secondary_VNIC_OCID>"
    ```

    c. When the script prompts you to overwrite the network interface's settings, say yes.

Windows bare metal instances: adding additional secondary VNICs

If you have one secondary VNIC on the second physical NIC of a bare metal instance, and you want to one or more additional VNICs, here's the overall process. It includes a task for setting up NIC teaming, which is required if the instance has more than one VNIC on the second physical NIC.



Note

If you increase the number of secondary VNICs on the second physical NIC from one to two or more, you must enable *NIC teaming* for the second physical NIC (see instructions that follow). In your NIC "team," you create a separate interface for *each* secondary VNIC on that physical NIC, including the initial one. This means that the original interface for that first secondary VNIC will no longer work, and any subsequent configuration you want to do for that VNIC must be done instead on the VNIC's new interface that's part of the "team".

1. Add one or more additional secondary VNICs to your instance. Keep each VNIC's OCID and VLAN tag handy so you can provide them when later running the configuration script. You can also get the values from the instance's VNIC metadata.

2. Connect to the instance with Remote Desktop.
3. Set up NIC teaming on the instance:
    a. Open the Server manager, and then click **Local Server**.
    b. In the list of properties, locate **NIC Teaming**, and then click **Disabled** to enable and set up NIC teaming.
    c. In the **Teams** section on the lower-left side of the screen, click **Tasks**, and then click **New Team**.
    d. Enter a name for the team, select the check box for the second physical NIC on the instance, and click **OK**.
    The new team is created and appears in the list of teams in the **Teams** section.
    e. Click the new team so it's selected, and then in the **Adapters and Interfaces** section on the right side of the screen, click the **Team Interfaces** tab.
    f. Click **Tasks**, and then click **Add Interface** (you will create a separate interface for each secondary VNIC on the second physical NIC).
    g. Click the radio button for **Specific VLAN**, and then enter the Oracle-assigned VLAN tag number for the VLAN (for example, 1). You can get the VLAN tag from the Console or the instance's VNIC metadata.
    h. Click **OK**.
    i. Repeat the four preceding steps (e-h) for each of the other secondary VNICs. You create a separate interface for each secondary VNIC.
4. Run the script:
    a. Open PowerShell as an administrator.
    b. For the first secondary VNIC, run the script with the secondary VNIC's OCID:

```
.\secondary_vnic_windows_configure.ps1 "<secondary_VNIC_OCID>"
```

    c. When the script prompts you to overwrite the network interface's settings, say yes.

d. Repeat the preceding two steps (b and c) for each of the additional secondary VNICs.

Private IP Addresses

This topic describes how to manage the IP addresses assigned to an instance in a virtual cloud network (VCN).



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Overview of IP Addresses

Instances use IP addresses for communication. Each instance has at least one private IP address and optionally one or more public IP addresses. A private IP address enables the instance to communicate with other instances inside the VCN, or with hosts in your on-premises network (via an IPSec VPN or Oracle Cloud Infrastructure FastConnect). A public IP address enables the instance to communicate with hosts on the internet. For more information, see these related topics:

- Public vs. Private Subnets
- How IP Addresses Are Assigned
- Public IP Addresses

About the Private IP Object

The Networking service defines an object called a *private IP*, which consists of:

- Private IPv4 address, assigned by either you or Oracle.
- Optional hostname for DNS (see DNS in Your Virtual Cloud Network).

Each private IP object has an Oracle-assigned OCID (see Resource Identifiers). If you're using the API, you can also assign each private IP object a friendly name.

Each instance receives a *primary private IP* object during launch. The Networking service uses the Dynamic Host Configuration Protocol (DHCP) to pass the object's private IP address to the instance. This address does not change during the instance's lifetime and cannot be removed from the instance. The private IP object is terminated when the instance is terminated.

If an instance has any secondary VNICs attached, each of those VNICs also has a primary private IP.

A private IP can have a public IP assigned to it at your discretion.

A private IP can be the target of a route rule in your VCN. For more information, see Using a Private IP as a Route Target.

About Secondary Private IP Addresses

You can add a *secondary private IP* to an instance after it's launched. You can add it to either the primary VNIC or a secondary VNIC on the instance. The secondary private IP address must come from the CIDR of the VNIC's subnet. You can move a secondary private IP from a VNIC on one instance to a VNIC on another instance if both VNICs belong to the same subnet.

Here are a few reasons why you might use secondary private IPs:

- **Instance failover:** You assign a secondary private IP to an instance. Then if the instance has problems, you can easily reassign that secondary private IP to a standby instance in the same subnet. If the secondary private IP has a public IP assigned to it, that public IP moves along with the private IP.

- **Running multiple services or endpoints on a single instance:** For example, you could have multiple container pods running on a single instance, and each uses an IP address from the VCN's CIDR. The containers have direct connectivity to other instances and services in the VCN. Another example: you could run multiple SSL websites with each one using its own IP address.

Here are more details about secondary private IP addresses:

- They're supported for all shapes and OS types, for both bare metal and VM instances.
- A VNIC can have a maximum of 31 secondary private IPs.
- They can be assigned only after the instance is launched (or the secondary VNIC is created/attached).
- A secondary private IP that is assigned to a VNIC in a regional subnet has a null availability domain attribute. Compare this with the VNIC's *primary* private IP, which always has its availability domain attribute set to the instance's availability domain, regardless of whether the instance's subnet is regional or AD-specific.
- Deleting a secondary private IP from a VNIC returns the address to the pool of available addresses in the subnet.
- They are automatically deleted when you terminate the instance (or detach/delete the secondary VNIC).
- The instance's bandwidth is fixed regardless of the number of private IP addresses attached. You can't specify a bandwidth limit for a particular IP address on an instance.
- A secondary private IP can have a reserved public IP assigned to it at your discretion.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: see IAM Policies for Networking.

Using the Console

To view an instance's private IPs

1. Confirm you're viewing the compartment that contains the instance you're interested in.
2. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
3. Click the instance to view its details.
4. Under **Resources**, click **Attached VNICs**.
   The primary VNIC and any secondary VNICs assigned to the instance are displayed.
5. Click the VNIC you're interested in.
6. Under **Resources**, click **IP Addresses**.
   The VNIC's primary private IP and any secondary private IPs are displayed.

To assign a new secondary private IP to a VNIC

1. Confirm you're viewing the compartment that contains the instance you're interested in.
2. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
3. Click the instance to view its details.
4. Under **Resources**, click **Attached VNICs**.
   The primary VNIC and any secondary VNICs attached to the instance are displayed.
5. Click the VNIC you're interested in.
6. Under **Resources**, click **IP Addresses**.
   The VNIC's primary private IP and any secondary private IPs are displayed.
7. Click **Assign Private IP Address**.
8. Enter the following:

- **Private IP Address:** Optional. An available private IP address of your choice from the subnet's CIDR (otherwise the private IP address is automatically assigned).
- **Unassign if already assigned to another VNIC:** Select this check box to force reassignment of the IP address if it's already assigned to another VNIC in the subnet. Relevant only if you specify a private IP address in the preceding field.
- **Hostname:** Optional. A hostname to be used for DNS within the cloud network. Available only if the VCN and subnet both have DNS labels. See DNS in Your Virtual Cloud Network.
- **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
- **Public IP address:** Whether to assign a public IP address. Available only if the VNIC is in a public subnet. See Public IP Addresses.

9. Click **Assign**.
   The secondary private IP is created and then displayed on the **IP Addresses** page for the VNIC.
10. Configure the IP address:
    - For instances running a variant of Linux, see Linux: Details about Secondary IP Addresses.
    - For Windows instances, see Windows: Details about Secondary IP Addresses.

To move a secondary private IP to another VNIC in the same subnet

1. Confirm you're viewing the compartment that contains the instance you're interested in.

2. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
3. Click the instance to view its details.
4. Under **Resources**, click **Attached VNICs**.
   The primary VNIC and any secondary VNICs attached to the instance are displayed.
5. Click the VNIC you're interested in.
6. Under **Resources**, click **IP Addresses**.
   The VNIC's primary private IP and any secondary private IPs are displayed.
7. Click **Assign Private IP Address**.
8. Enter the following:
   - **Private IP Address:** The secondary private IP address you want to move.
   - **Unassign if already assigned to another VNIC:** Select this check box to move the secondary IP address from the VNIC it's currently assigned to.
   - **Hostname:** Optional. The hostname to be used for DNS within the cloud network. Available only if the VCN and subnet both have DNS labels. See DNS in Your Virtual Cloud Network.
   - **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
   - **Public IP address:** Whether to assign a public IP address. Available only if the VNIC is in a public subnet. See Public IP Addresses.
9. Click **Assign**.

The private IP address is moved from the original VNIC to the new VNIC.

To update the hostname for an existing private IP

1. Confirm you're viewing the compartment that contains the instance you're interested in.
2. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
3. Click the instance to view its details.
4. Under **Resources**, click **Attached VNICs**.
   The primary VNIC and any secondary VNICs attached to the instance are displayed.
5. Click the VNIC you're interested in.
6. Under **Resources**, click **IP Addresses**.
   The VNIC's primary private IP and any secondary private IPs are displayed.
7. For the IP address you're interested in, click the Actions icon (three dots), and then click **Edit**.
8. Make your changes and click **Update**.

To delete a secondary private IP from a VNIC



Warning

If the private IP is the target of a route rule, deleting it from the VNIC causes the route rule to blackhole and the traffic will be dropped.

Prerequisite: Oracle recommends removing the IP address from the OS configuration before deleting it from the VNIC. See Linux: Details about Secondary IP Addresses or Windows: Details about Secondary IP Addresses.

1. Confirm you're viewing the compartment that contains the instance you're interested in.
2. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click

Instances.

3. Click the instance to view its details.
4. Under **Resources**, click **Attached VNICs**.
   The primary VNIC and any secondary VNICs attached to the instance are displayed.
5. Click the VNIC you're interested in.
6. Under **Resources**, click **IP Addresses**.
   The VNIC's primary private IP and any secondary private IPs are displayed.
7. For the private IP you want to delete, click the Actions icon (three dots), and then click **Delete Private IP**.
8. Confirm when prompted.

The private IP address is returned to the pool of available addresses in the subnet.

To manage tags for a private IP

1. Confirm you're viewing the compartment that contains the instance you're interested in.
2. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
3. Click the instance to view its details.
4. Under **Resources**, click **Attached VNICs**.
   The primary VNIC and any secondary VNICs attached to the instance are displayed.
5. Click the VNIC you're interested in.
6. Under **Resources**, click **IP Addresses**.
   The VNIC's primary private IP and any secondary private IPs are displayed.
7. For the private IP you're interested in, click the Actions icon (three dots), and then click **View Tags**. From there you can view the existing tags, edit them, and apply new ones.

For more information, see Resource Tags.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

To manage private IPs on a VNIC, use these operations:

- GetPrivateIp: Use this to get a single `privateIp` object by specifying its OCID.
- ListPrivateIps: Use this to get a single `privateIp` object by specifying the private IP address (for example, 10.0.3.3) and the subnet's OCID. Or you can list all the `privateIp` objects in a given subnet, or just the ones assigned to a given VNIC.
- CreatePrivateIp: Use this to assign a new secondary private IP to a VNIC.
- UpdatePrivateIp: Use this to reassign a secondary private IP to a different VNIC in the same subnet, or to update the hostname or display name of a secondary private IP.
- DeletePrivateIp: Use this to delete a secondary private IP from a VNIC. The private IP address is returned to the subnet's pool of available addresses.

Linux: Details about Secondary IP Addresses

After assigning a secondary private IP to a VNIC, you must configure the OS to use it.

Basic Commands (Not Persistent Through a Reboot)

On the instance, run the following command. It works on all variants of Linux, for both bare metal and VM instances:

```
ip addr add <address>/<subnet_prefix_len> dev <phys_dev> label <phys_dev>:<addr_seq_num>
```

where:

- `<address>`: The secondary private IP address.
- `<subnet_prefix_len>`: The subnet's prefix length. For example, if the subnet is 192.168.20.0/24, the subnet prefix length is 24.
- `<phys_dev>`: The interface to add the address to (for example, ens2f0).

- <addr_seq_num>: The sequential number in the stack of addresses on the device (for example, 0).

For example:

```
ip addr add 192.168.20.50/24 dev ens2f0 label ens2f0:0
```

Later if you want to delete the address, you can use:

```
ip addr del 192.168.20.50/24 dev ens2f0:0
```

Also make sure to delete the secondary IP from the VNIC. You can do that before or after executing the above command to delete the address from the OS configuration.



Note

If you've assigned a secondary IP to a secondary VNIC, and you're using policy-based routing for the secondary VNIC, make sure to configure the route rules to look up the same route table for the secondary IP address.

Configuration File (Persistent Through a Reboot)

You can make the configuration persistent through a reboot by adding the information to a configuration file.

For Oracle Linux and CentOS

Create an `ifcfg` file named `/etc/sysconfig/network-scripts/ifcfg-<phys_dev>:<addr_seq_num>`. To continue with the preceding example, the file name would be `/etc/sysconfig/network-scripts/ifcfg-ens2f0:0`, and the contents would be:

```
DEVICE="ens2f0:0"
BOOTPROTO=static
IPADDR=192.168.20.50
NETMASK=255.255.255.0
ONBOOT=yes
```



Note

If you've assigned a secondary IP to a secondary VNIC, and you're using policy-based routing for the secondary VNIC, make sure to configure the route rules to look up the same route table for the secondary IP address.

For Ubuntu

Add the following information to the end of the `/etc/network/interfaces` file:

```
auto <phys_dev>:<addr_seq_num>
iface <phys_dev>:<addr_seq_num> inet static
       address <address>
       netmask <address_netmask>
```

Where the netmask is not the prefix but the 255.255.x.x. address.

To continue with the earlier example:

```
auto ens2f0:0
iface ens2f0:0 inet static
       address 192.168.20.50
       netmask 255.255.255.0
```



Note

If you've assigned a secondary IP to a secondary VNIC, and you're using policy-based routing for the secondary VNIC, make sure to configure the route rules to look up the same route table for the secondary IP address.

Windows: Details about Secondary IP Addresses

After assigning a secondary private IP to a VNIC, you must configure the OS to use it. Here are instructions for using a PowerShell script or the Network and Sharing Center UI.

Using a PowerShell Script

You must run PowerShell as an administrator. The script configures two things: static IP addressing on the instance and the secondary private IP. The configuration persists through a reboot of the instance.

1. In your browser, go to the Console, and note the secondary private IP address that you want to configure on the instance.
2. Connect to the instance, and run the following command at a command prompt:

   ```
   ipconfig /all
   ```

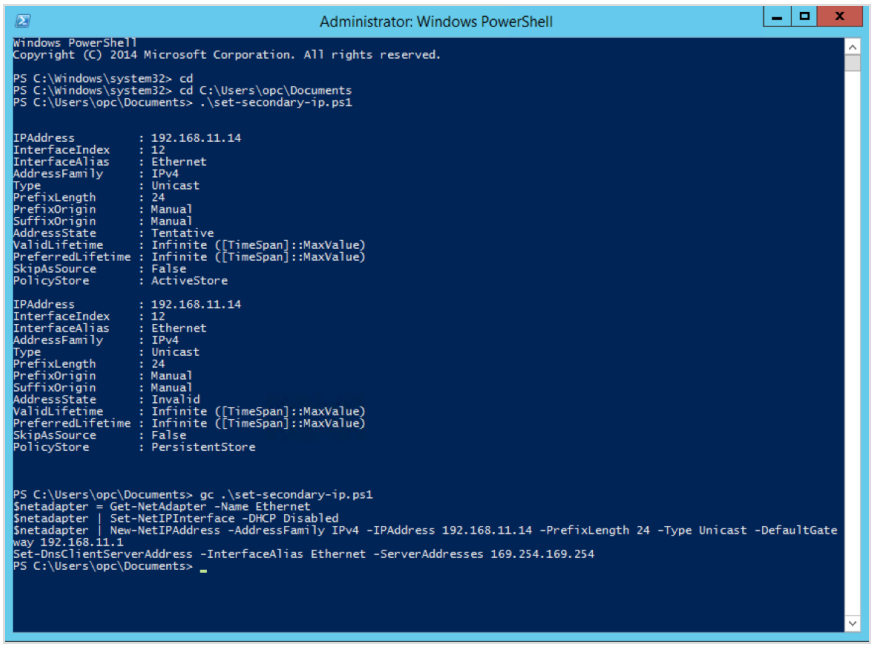
3. Note the values for the following items so you can enter them into the script in the next step:
   - Default Gateway
   - DNS Servers
4. Replace the variables in the following PowerShell script with your own values:

   ```
   netadapter = Get-NetAdapter -Name Ethernet
   $netadapter | Set-NetIPInterface -DHCP Disabled
   $netadapter | New-NetIPAddress -AddressFamily IPv4 -IPAddress <secondary_IP_address> -
   PrefixLength <subnet_prefix_length> -Type Unicast -DefaultGateway <default_gateway>
   Set-DnsClientServerAddress -InterfaceAlias Ethernet -ServerAddresses <DNS_server>
   ```

   For example:

   ```
   netadapter = Get-NetAdapter -Name Ethernet
   $netadapter | Set-NetIPInterface -DHCP Disabled
   $netadapter | New-NetIPAddress -AddressFamily IPv4 -IPAddress 192.168.11.14 -PrefixLength 24 -
   Type Unicast -DefaultGateway 192.168.11.1
   Set-DnsClientServerAddress -InterfaceAlias Ethernet -ServerAddresses 169.254.169.254
   ```

5. Save the script with the name of your choice and a `.ps1` extension, and run it on the instance.



If you run `ipconfig /all` again, you'll see that DHCP has been disabled and the secondary private IP address is included in the list of IP addresses.

Later if you want to delete the address, you can use this command:

```
Remove-NetIPAddress -IPAddress 192.168.11.14 -InterfaceAlias Ethernet
```

Also make sure to delete the secondary IP from the VNIC. You can do that before or after executing the above command to delete the address from the OS configuration.

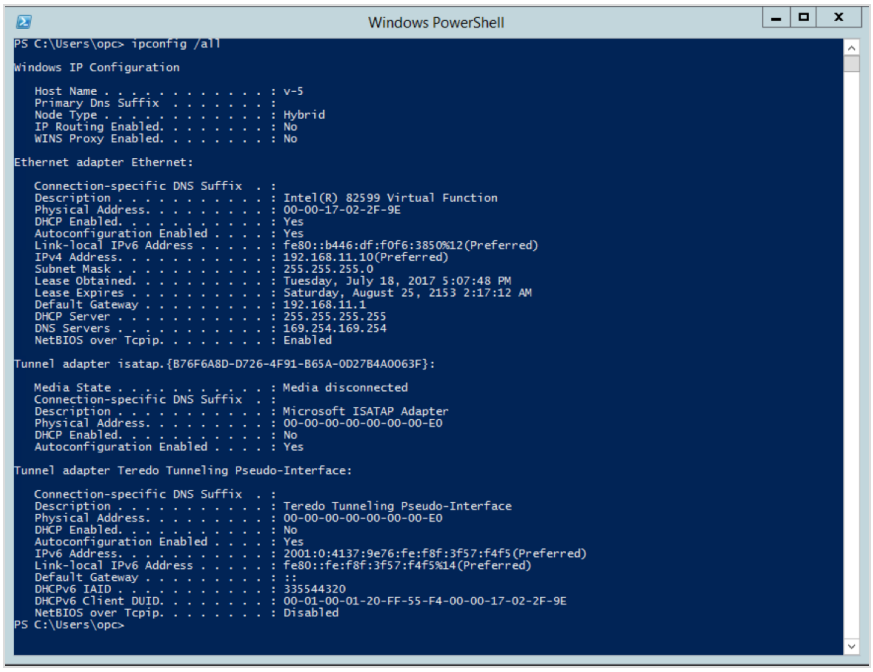
Using the Network and Sharing Center UI

The following instructions configure two things: static IP addressing on the instance and the secondary private IP. The configuration persists through a reboot of the instance.

1. In your browser, go to the Console, and note the secondary private IP address that you want to configure on the instance.

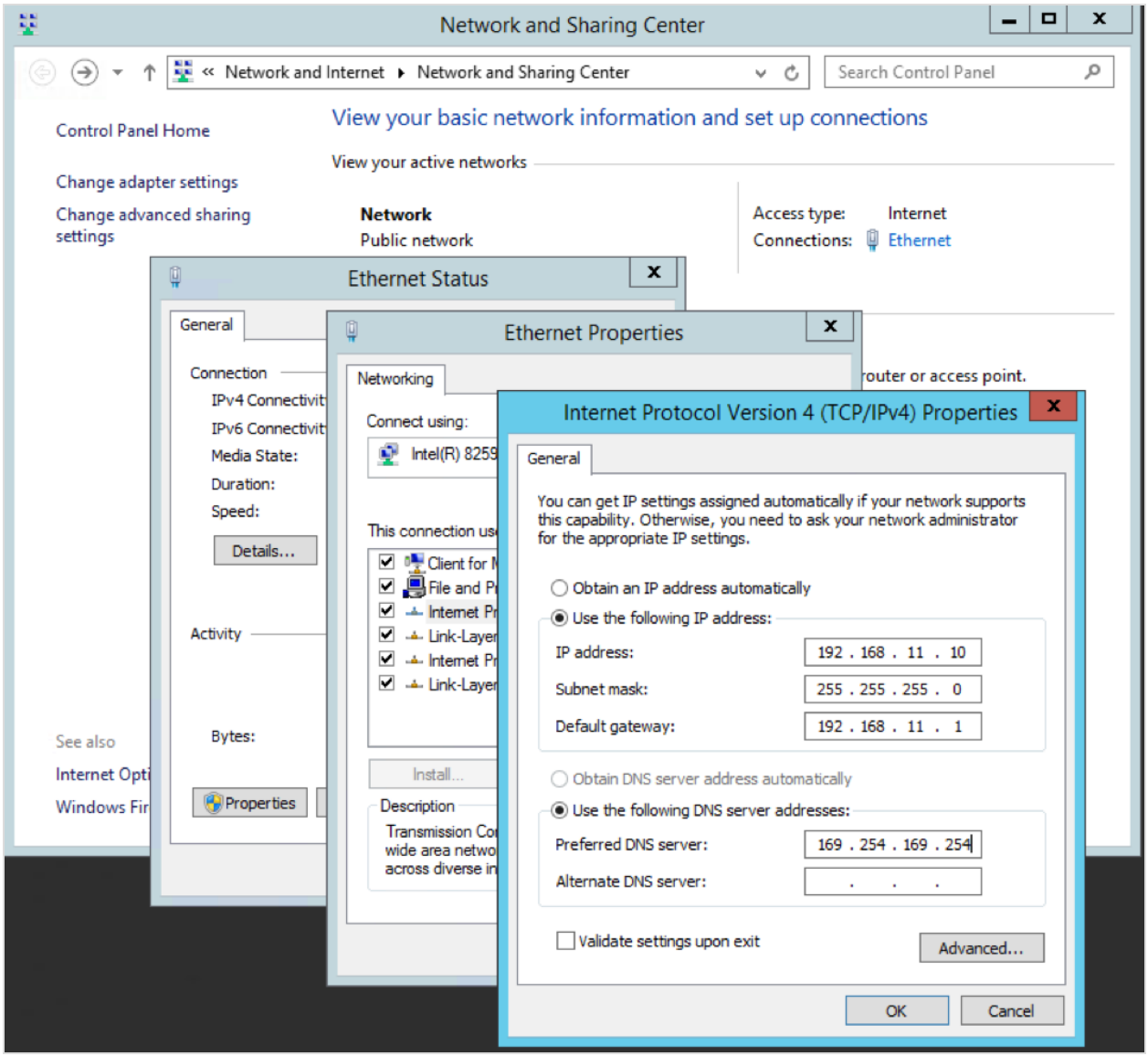
2. Connect to the instance, and run the following command at a command prompt:

```
ipconfig /all
```



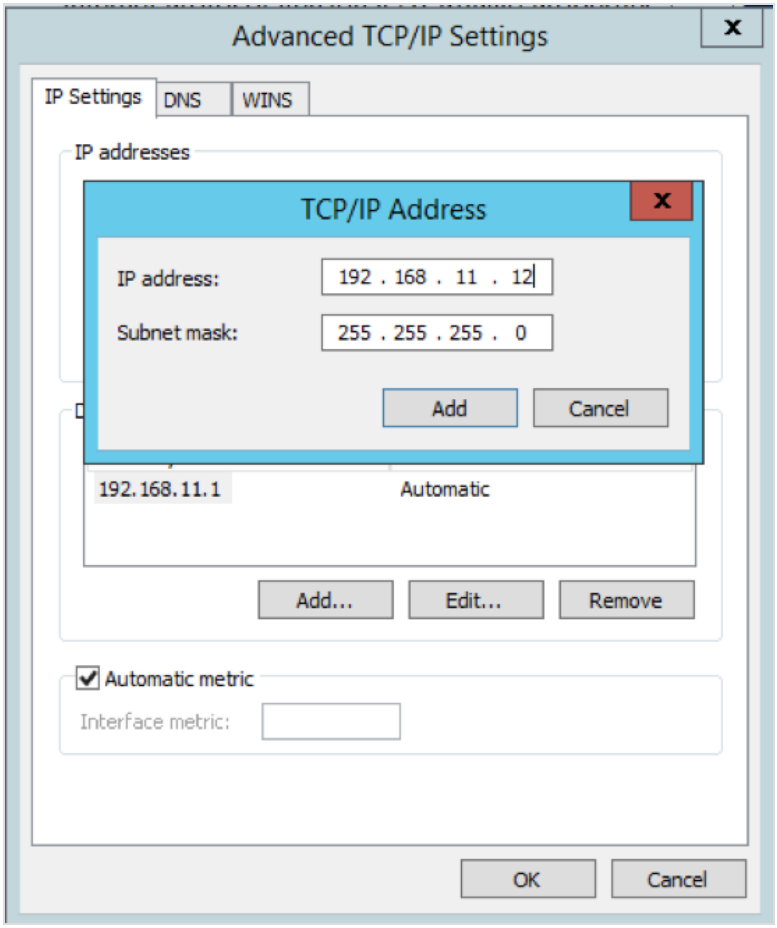
3. Note the values for the following items so you can enter them elsewhere in a later step:
    - IPv4 Address
    - Subnet Mask
    - Default Gateway
    - DNS Servers
4. In the instance's **Control Panel**, open the **Network and Sharing Center** (see the image below for the set of dialog boxes you'll see in these steps).
5. For the active networks, click the connection (**Ethernet**).

6. Click **Properties**.
7. Click **Internet Protocol Version 4 (TCP/IPv4)**, and then click **Properties**.
8. Select the radio button for **Use the following IP address**, and then enter the values you noted earlier for the IP address, subnet mask, default gateway, and DNS servers.



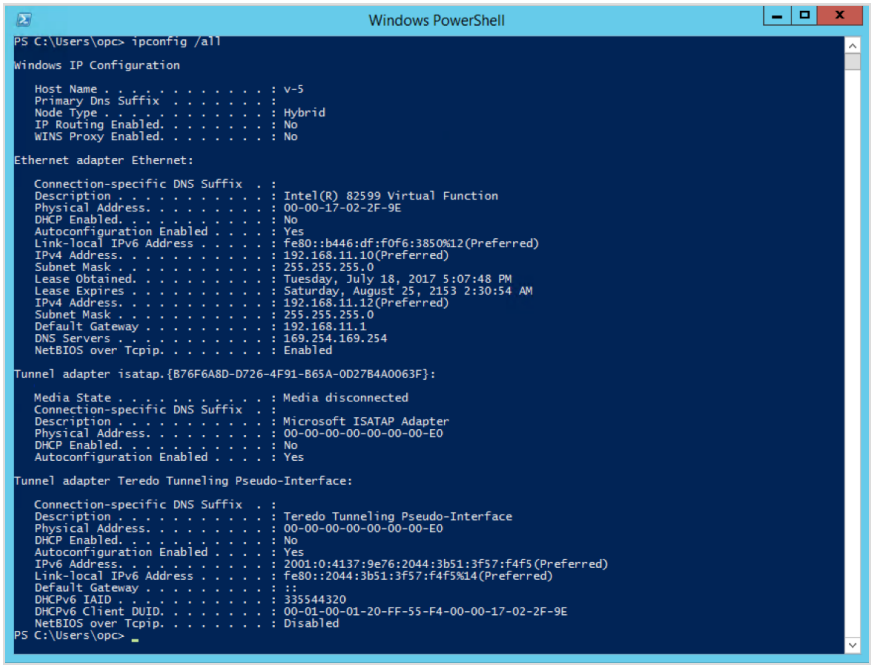
9. Click **Advanced...**.
10. Under **IP addresses**, click **Add...**.

11. Enter the secondary private IP address and the subnet mask you used earlier and click **Add**.



12. Click **OK** until the Network and Sharing Center is closed.
13. Verify the changes by returning to the command prompt and running `ipconfig /all`.

You should now see that DHCP is disabled (static IP addressing is enabled), and the secondary private IP address is in the list of addresses displayed. The address is now configured on the instance and available to use.



Note

You might not see the primary private IP address when you again view the properties for Internet Protocol Version 4 (TCP/IPv4) in the Network and Sharing Center UI. The best way to confirm your changes is to use `ipconfig /all` at the command line.

Public IP Addresses

This topic describes how to manage public IP addresses on instances in a virtual cloud network (VCN).



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Overview of Public IP Addresses

A public IP address is an IPv4 address that is reachable from the internet. If a resource in your tenancy needs to be directly reachable from the internet, it must have a public IP address. Depending on the type of resource, there might be other requirements.

Certain types of resources in your tenancy are designed to be directly reachable from the internet and therefore automatically come with a public IP address. For example: a NAT gateway or a public load balancer. Other types of resources are directly reachable only if you configure them to be. For example: instances in your VCN.

This topic focuses on these subjects:

- The types of public IP addresses and their characteristics
- How to control whether an instance has a public IP address

For more information about resources that automatically get a public IP address, see Resources That Always Get a Public IP.

Instances and Public IP Addresses

You can assign a public IP address to an instance to enable communication with the internet.

The instance is assigned a public IP address from the Oracle Cloud Infrastructure address pool.

The assignment is actually to a private IP object on the instance. The VNIC that the private IP is assigned to must be in a public subnet. A given instance can have multiple secondary VNICs, and a given VNIC can have multiple secondary private IPs. So you can assign a given instance multiple public IPs across one or more VNICs if you like.

For an instance to communicate directly with the internet, all of the following are required:

- The instance must be in a public subnet.
- The instance must have a public IP address.
- The instance's VCN must have an internet gateway.
- The public subnet must have route tables and security lists configured accordingly.



Tip

Oracle Cloud Infrastructure FastConnect public peering lets your on-premises network access the public IP addresses of resources in Oracle Cloud Infrastructure *without the traffic traversing the internet*. For more information, see FastConnect.

The Public IP Object

The Networking service defines an object called a *public IP*, which consists of these items:

- Public IPv4 address (chosen by Oracle)
- Properties that further define the public IP's type and behavior

Each public IP object has an Oracle-assigned OCID (see Resource Identifiers). If you're using the API, you can also assign each public IP object a friendly name.

Types of Public IPs

There are two types of public IPs:

- **Ephemeral:** Think of it as temporary and existing for the lifetime of the instance.
- **Reserved:** Think of it as persistent and existing beyond the lifetime of the instance it's assigned to. You can unassign it and then reassign it to another instance whenever you like. Exception: reserved public IPs on public load balancers. See Resources That Always Get a Public IP.

The following table summarizes the differences between the two types.

Characteristic	Ephemeral Public IPs	Reserved Public IPs
Allowed assignment	To a VNIC's primary private IP only Limits: • One per VNIC • Two per VM instance, and 16 per bare metal instance	To either a primary or secondary private IP Limit: 32 per VNIC
Creation	Optionally created and assigned during instance launch or secondary VNIC creation. You can create and assign one later if the VNIC doesn't already have one.	You create one at any time. You can then assign it when you like. Limit: You can create 50 per region

Characteristic	Ephemeral Public IPs	Reserved Public IPs
Unassignment	You can unassign it at any time, which deletes it. You might do this if whoever launched the instance included a public IP, but you don't want the instance to have one. When you stop an instance, its ephemeral public IPs remain assigned to the instance.	You can unassign it at any time, which returns it to your tenancy's pool of reserved public IPs.
Moving to a different resource	You cannot move an ephemeral public IP to a different private IP.	If assigned to a secondary private IP: If you move the private IP to a different VNIC (must be in the same subnet), the reserved public IP goes with it. You can move it (unassign and then reassign it) at any time to another private IP in the same region. Can be in a different VCN or availability domain.
Automatic deletion	Its lifetime is tied to the private IP's lifetime. Automatically unassigned and deleted when: • Its private IP is deleted • Its VNIC is detached or terminated • Its instance is terminated	Never. Exists until you delete it.

Characteristic	Ephemeral Public IPs	Reserved Public IPs
Scope	Availability domain	Regional (can be assigned to a private IP in any availability domain in the region)
Compartment and availability domain	Same as the private IP's	Can be different from the private IP's

When you launch an instance in a public subnet, by default, the instance gets a public IP unless you say otherwise. See To choose whether an ephemeral public IP is assigned when launching an instance.

After you create a given public IP, you can't change which type it is. For example, if you launch an instance that is assigned an ephemeral public IP with address 129.146.1.9, you can't convert the ephemeral public IP to a reserved public IP with address 129.146.1.9.

The preceding table notes the public IP limits per VNIC and instance. If you try to perform any operation that assigns or moves a public IP to a VNIC or instance that has already reached its public IP limit, an error is returned. The operations include:

- Assigning a public IP
- Creating a new secondary VNIC with a public IP
- Moving a private IP with a public IP to another VNIC
- Moving a public IP to another private IP

Resources That Always Get a Public IP

As mentioned earlier, certain types of resources are designed to be directly reachable from the internet. Examples: a NAT gateway or a public load balancer. These resources automatically get a public IP address upon creation. Oracle chooses the public IP address from the Oracle pool. You can't remove or change the address.

For public load balancers, the address is a regional reserved public IP that is assigned to a private IP on the load balancer. This public IP appears in the list of your tenancy's reserved public IPs, which you can view in the Console. However, unlike other reserved public IPs that you create, you have no control over this public IP. You can't edit it or unassign it from the load balancer yourself. It's automatically unassigned and deleted from your tenancy when you terminate the load balancer.

For NAT gateways, the address is a regional ephemeral public IP that is assigned to the NAT gateway. Like other ephemeral public IPs, it's automatically unassigned and deleted when you terminate its assigned resource (the NAT gateway). However, unlike other ephemeral public IPs, you can't edit it or unassign it yourself.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: see IAM Policies for Networking.

Ephemeral Public IPs: Using the Console

To choose whether an ephemeral public IP is assigned when launching an instance

When you launch an instance into a public subnet, there's an **Assign public IP address** check box in the **Launch Instance** dialog box (in the **Advanced Options**, on the **Networking** tab). By default, the check box is selected, which means the instance gets an ephemeral public IP.

If you do NOT want an ephemeral public IP assigned, you can either:

- Clear the **Assign public IP address** check box
- Delete the ephemeral public IP after instance launch

To assign an ephemeral public IP when creating a secondary VNIC

When you add a secondary VNIC to an instance, you choose whether the primary private IP on the new VNIC gets an ephemeral public IP. This choice is available only if the secondary VNIC is in a public subnet.

In the **Create VNIC** dialog box, there's an **Assign public IP address** check box. By default, the check box is NOT selected, which means the secondary VNIC does not get an ephemeral public IP. You must select the check box.

For instructions, see To create and attach a secondary VNIC.

To assign an ephemeral public IP to an existing primary private IP

Prerequisite: The primary private IP must not have a reserved or ephemeral public IP already assigned to it. If it does, first delete the ephemeral public IP, or unassign the reserved public IP.

1. Confirm you're viewing the compartment that contains the instance you're interested in.
2. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
3. Click the instance to view its details.
4. Under **Resources**, click **Attached VNICs**.
   The primary VNIC and any secondary VNICs attached to the instance are displayed.
5. Click the VNIC you're interested in.
6. Under **Resources**, click **IP Addresses**.
   The VNIC's primary private IP and any secondary private IPs are displayed.
7. For the VNIC's primary private IP, click the Actions icon (three dots), and then click **Edit**.

8. In the **Public IP Address** section, for **Public IP type**, select the radio button for **Ephemeral Public IP**.
9. In the **Ephemeral Public IP Name** field, enter an optional friendly name for the public IP. The name doesn't have to be unique, and you can change it later. Avoid entering confidential information.
10. Click **Update**.

To delete an ephemeral public IP from an instance

Deleting an ephemeral public IP automatically unassigns it from its private IP.

1. Confirm you're viewing the compartment that contains the instance you're interested in.
2. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
3. Click the instance to view its details.
4. Under **Resources**, click **Attached VNICs**.
   The primary VNIC and any secondary VNICs attached to the instance are displayed.
5. Click the VNIC you're interested in.
6. Under **Resources**, click **IP Addresses**.
   The VNIC's primary private IP and any secondary private IPs are displayed.
7. For the VNIC's primary private IP, click the Actions icon (three dots), and then click **Edit**.
8. In the **Public IP Address** section, for **Public IP Type**, select the radio button for **No Public IP**.
9. Click **Update**.

To change the display name for an ephemeral public IP

1. Confirm you're viewing the compartment that contains the instance you're interested in.

2. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
3. Click the instance to view its details.
4. Under **Resources**, click **Attached VNICs**.
   The primary VNIC and any secondary VNICs attached to the instance are displayed.
5. Click the VNIC you're interested in.
6. Under **Resources**, click **IP Addresses**.
   The VNIC's primary private IP and any secondary private IPs are displayed.
7. For the VNIC's primary private IP, click the Actions icon (three dots), and then click **Edit**.
8. In the **Public IP Address** section, edit the **Ephemeral Public IP Name**. The name doesn't have to be unique, and you can change it later. Avoid entering confidential information.
9. Click **Update**.

Reserved Public IPs: Using the Console

To view your reserved public IPs

1. Confirm you're viewing the region and compartment you're interested in.
2. Open the navigation menu. Under Core Infrastructure, go to **Networking** and click **Public IPs**.

The details of the reserved public IPs in the selected region and compartment are displayed. If the reserved public IP is assigned, there's a link to the relevant VNIC.

To create a new reserved public IP in your pool

1. Confirm you're viewing the region and compartment where you want to create the

reserved public IP.

2. Open the navigation menu. Under Core Infrastructure, go to **Networking** and click **Public IPs**.
3. Click **Create Reserved Public IP**.
4. Enter the following:
   - **Name**: An optional friendly name for the reserved public IP. The name doesn't have to be unique, and you can change it later. Avoid entering confidential information.
   - **Compartment:** Leave as is.
   - **Tags:**Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
5. Click **Create Reserved Public IP**.

The new reserved public IP is created and displayed on the page. You can now assign it to an existing private IP if you like.

To delete a reserved public IP from your pool

The reserved public IP can be in the "Assigned" state. Deleting a reserved public IP automatically unassigns it from its private IP.

1. Confirm you're viewing the region and compartment that contains the reserved public IP.
2. Open the navigation menu. Under Core Infrastructure, go to **Networking** and click **Public IPs**.

3. For the reserved public IP you want to delete, click the Actions icon (three dots), and then click **Terminate**.
4. Confirm when prompted.

After a few seconds, the reserved public IP is unassigned (if it was assigned) and deleted from your pool.

To assign a reserved public IP to a private IP

Prerequisite: The private IP must not have an ephemeral or reserved public IP already assigned to it. If it does, first delete the ephemeral public IP, or unassign the reserved public IP.

1. Confirm you're viewing the compartment that contains the instance with the private IP you're interested in.
2. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
3. Click the instance to view its details.
4. Under **Resources**, click **Attached VNICs**.
   The primary VNIC and any secondary VNICs attached to the instance are displayed.
5. Click the VNIC you're interested in.
6. Under **Resources**, click **IP Addresses**.
   The VNIC's primary private IP and any secondary private IPs are displayed.
7. For the private IP you're interested in, click the Actions icon (three dots), and then click **Edit**.
8. In the **Public IP Address** section, for **Public IP Type**, select the radio button for **Reserved Public IP**.
9. Enter the following:
   - **Compartment:** The compartment that contains the reserved public IP you want to assign.

- **Reserved Public IP:** The reserved public IP you want to assign. You have three choices:
  - Create a new reserved public IP. You may optionally provide a friendly name for it. The name doesn't have to be unique, and you can change it later. Avoid entering confidential information.
  - Assign a reserved public IP that is currently unassigned.
  - Move a reserved public IP from another private IP.

10. Click **Update**.

To unassign a reserved public IP and return it to the pool

1. Confirm you're viewing the compartment that contains the instance with the reserved public IP you're interested in.
2. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
3. Click the instance to view its details.
4. Under **Resources**, click **Attached VNICs**.
   The primary VNIC and any secondary VNICs attached to the instance are displayed.
5. Click the VNIC you're interested in.
6. Under **Resources**, click **IP Addresses**.
   The VNIC's primary private IP and any secondary private IPs are displayed.
7. For the private IP you're interested in, click the Actions icon (three dots), and then click **Edit**.
8. In the **Public IP Address** section, for **Public IP Type**, select the radio button for **No Public IP**.
9. Click **Update**.

The reserved public IP is unassigned and returned to your pool.

To move a reserved public IP from one private IP to another

Let's say you want to move a reserved public IP from private IP 1 to private IP 2. In summary: Make sure private IP 2 doesn't have a public IP already assigned to it. Then assign the reserved public IP to private IP 2. It will be automatically unassigned from private IP 1 first. Detailed instructions:

1. Confirm you're viewing the compartment that contains the instance with private IP 2.
2. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
3. Click the instance to view its details.
4. Under **Resources**, click **Attached VNICs**.
   The primary VNIC and any secondary VNICs attached to the instance are displayed.
5. Click the VNIC you're interested in.
6. Under **Resources**, click **IP Addresses**.
   The VNIC's primary private IP and any secondary private IPs are displayed.
7. For private IP 2, click the Actions icon (three dots), and then click **Edit**.
8. If private IP 2 already has a public IP assigned to it:
   a. In the **Public IP Address** section, select the radio button for **No Public IP**.
   b. Click **Update**.
   c. Again for private IP 2, click the Actions icon (three dots), and then click **Edit**.
9. In the **Public IP Address** section, select the radio button for **Reserved Public IP**.
10. Enter the following:
    - **Compartment:** The compartment that contains the reserved public IP you want to assign.
    - **Reserved Public IP:** The reserved public IP you want to assign. It will be moved from the public IP it's currently assigned to.
11. Click **Update**.

To change the display name for a reserved public IP

1. Confirm you're viewing the region and compartment that contains the reserved public IP.
2. Open the navigation menu. Under Core Infrastructure, go to **Networking** and click **Public IPs**.
3. For the reserved public IP you want to edit, click the Actions icon (three dots), and then click **Edit**.
4. Make your changes to the friendly name. The name doesn't have to be unique, and you can change it later. Avoid entering confidential information.
5. Click **Save**.

To manage tags for a reserved public IP

1. Confirm you're viewing the region and compartment that contains the reserved public IP.
2. Open the navigation menu. Under Core Infrastructure, go to **Networking** and click **Public IPs**.
3. For the reserved public IP you're interested in, click the Actions icon (three dots), and then click **View Tags**. From there you can view the existing tags, edit them, and apply new ones.

For more information, see Resource Tags.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

To manage public IPs, use these operations:

- GetPublicIp: Use this to get a `publicIp` object by specifying its OCID.
- GetPublicIpByIpAddress: Use this to get a `publicIp` object by specifying its public IP address.
- GetPublicIpByPrivateIpId: Use this to get a `publicIp` object by specifying the OCID of the private IP it's assigned to.
- ListPublicIps: Use this to list either your ephemeral or reserved `publicIp` objects.
- CreatePublicIp: Use this to create a new reserved public IP in your pool.
- UpdatePublicIp: Use this to assign, reassign, or unassign a reserved public IP, or to update the display name of an ephemeral or reserved public IP. You can also update a reserved public IP's tags.
- DeletePublicIp: Use this to delete an ephemeral public IP from its private IP, or delete a reserved public IP from your pool. The operation first unassigns the public IP.

DNS in Your Virtual Cloud Network

The Domain Name System (DNS) lets computers use hostnames instead of IP addresses to communicate with each other.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Choices for DNS in Your VCN

Following are the choices for DNS name resolution for the instances in your VCN. You make this choice for *each subnet* in the VCN, using the subnet's set of DHCP options. This is similar to how you configure which route table and security lists are associated with each subnet. For more information, see DHCP Options.



Note

You use the Domain Name Server DHCP option to specify the DNS Type for the associated subnet. If you change the option's value, either restart the DHCP client on the instance or reboot the instance. Otherwise, the change does not get picked up until the DHCP client refreshes the lease (within 24 hours).

DEFAULT CHOICE: INTERNET AND VCN RESOLVER

This is an Oracle-provided option that includes two parts:

- **Internet Resolver:** Lets instances resolve hostnames that are publicly published on the internet. The instances do not need to have internet access by way of either an internet gateway or a connection to your on-premises network (such as an IPSec VPN connection through a DRG).
- **VCN Resolver:** Lets instances resolve hostnames (which you can assign) of other instances in the same VCN. For more information, see About the DNS Domains and Hostnames.

By default, new VCNs you create use the Internet and VCN Resolver. If you're using the Networking API, this choice refers to the `VcnLocalPlusInternet` enum in the DhcpDnsOption object.



Note

The Internet and VCN Resolver does not let instances resolve the hostnames of hosts in your on-premises network connected to your VCN by IPSec VPN connection or FastConnect. Use your own custom DNS resolver to enable that.

CUSTOM RESOLVER

Use DNS servers of your choice for resolution (maximum three). They could be DNS servers that are:

- Available through the internet. For example, 216.146.35.35 for Dyn's Internet Guide.
- In your VCN.
- In your on-premises network, which is connected to your VCN by way of an IPSec VPN connection or FastConnect (through a DRG).

About the DNS Domains and Hostnames

When you initially create a VCN and subnets, you may specify DNS labels for each. The labels, along with the parent domain of `oraclevcn.com` form the VCN domain name and subnet domain name:

- **VCN domain name:** *<VCN DNS label>*`.oraclevcn.com`
- **Subnet domain name:** *<subnet DNS label>*.*<VCN DNS label>*`.oraclevcn.com`

When you then launch an instance, you may assign a hostname. It's assigned to the VNIC that's automatically created during instance launch (that is, the *primary VNIC*). Along with the subnet domain name, the hostname forms the instance's fully qualified domain name (FQDN):

- **Instance FQDN:** *<hostname>*.*<subnet DNS label>*.*<VCN DNS label>*`.oraclevcn.com`

For example: `database1.privatesubnet1.abccorpvcn1.oraclevcn.com`.

The FQDN resolves to the instance's private IP address. The Internet and VCN Resolver also enables reverse DNS lookup, which lets you determine the hostname corresponding to the private IP address.

If you add a secondary VNIC to the instance, you can specify a hostname. The resulting FQDN resolves to the VNIC's private IP address (that is, the *primary private IP*).

If you add a secondary private IP to a VNIC, you can specify a hostname. The resulting FQDN resolves to that private IP address.

Requirements for DNS Labels and Hostnames

- VCN and subnet labels: Max 15 alphanumeric characters and must start with a letter. Note that hyphens are NOT allowed. The value cannot be changed later.
- Hostnames: Max 63 characters and must be compliant with RFCs 952 and 1123. The value can be changed later.



Important

The Networking service allows hostnames up to 63 characters. However, some older operating systems enforce shorter hostnames. In Linux, here's how to determine the maximum allowed hostname length:

```
getconf HOST_NAME_MAX
```

If an instance has a hostname longer than the OS-specific maximum, the instance's FQDN is not resolvable within the VCN. You can use the Networking service to update the VNIC and change the hostname to a shorter value.

Uniqueness:

- VCN DNS label should be unique across your VCNs (not required, but a best practice)
- Subnet DNS labels must be unique within the VCN
- Hostnames must be unique within the subnet



Tip

Don't confuse the DNS label or hostname with the friendly name you can assign to the object (that is, the *display name*), which doesn't have to be unique.

Validation and Generation of the Hostname

If you've set DNS labels for the VCN and subnets, Oracle validates the hostname for DNS compliance and uniqueness during instance launch. If either of these requirements isn't met, the launch request fails.

If you don't specify a hostname during instance launch, Oracle tries to use the instance's display name as the hostname. If the display name does not pass the validation, Oracle automatically generates a DNS-compliant hostname that's unique across the subnet. You can see the generated hostname on the instance's page in the Console. In the API, the hostname is part of the VNIC object.

If you don't provide a hostname or display name during instance launch, Oracle automatically generates a display name but NOT a hostname. This means the instance won't be resolvable using the Internet and VCN Resolver.



Note

The Linux OS hostname on the instance is automatically set to the hostname you set during instance launch (or the one generated by Oracle). If you change the hostname directly on the instance, the FQDN of the instance does not get updated.

If you add a secondary VNIC to an instance, or add a secondary private IP to a VNIC, Oracle never tries to generate a hostname. Provide a valid hostname if you want the private IP address to be resolvable using the Internet and VCN Resolver.

DHCP Options for DNS

There are two DHCP options related to DNS in your VCN:

- **Domain Name Server:** To specify your choice for DNS type (either Internet and VCN Resolver, or Custom Resolver).

    - **Default value in the default set of DHCP options:** Internet and VCN Resolver
- **Search Domain:** To specify a single search domain. When resolving a DNS query, the OS appends this search domain to the value being queried. You can specify only one search domain for the set of DHCP options.
    - **Default value in the default set of DHCP options:** The VCN domain name (*<VCN DNS label>*.oraclevcn.com), if you specified a DNS label for the VCN during creation but did not specify a search domain value. If you specified a search domain value, then that value is used for the Search Domain option. If you did NOT specify a DNS label, the default set of DHCP options does not include a Search Domain option.

The default set of DHCP options that you can associate with all the subnets in the VCN automatically uses the default values. This means you can simply use the *<hostname>*.*<subnet DNS label>* to communicate with any of the instances in the VCN. If the VCN uses a set of DHCP options that does not contain a Search Domain option, the instances must use the entire FQDN to communicate.



Important

In general, when *any* set of DHCP options is initially created (the default set or a custom set you create), the Networking service automatically adds the Search Domain option and sets it to the VCN domain name (`<VCN DNS label>.oraclevcn.com`) *if all of these are true*:

- The VCN has a DNS label
- DNS Type = Internet and VCN Resolver
- You did NOT specify a search domain of your choice during creation of the set of DHCP options

After the set of DHCP options is created, you can always remove the Search Domain option or set it to a different value.

How to Enable DNS Hostnames in Your VCN

Only new VCNs created after the release of the Internet and VCN Resolver feature have automatic access to it. How to enable DNS hostnames for a new VCN depends on which interface you're using.

If you create a new VCN and subnets with the Console

1. When creating the VCN:
   - Check the checkbox for **Use DNS Hostnames in this VCN**
   - Specify a DNS label of your choice for the VCN. If you check the checkbox but don't specify a DNS label, the Console assumes that you want to use the Internet and VCN Resolver in your VCN and automatically generates a DNS label for the

VCN. The Console takes the VCN name you provided, removes non-alphanumeric characters, ensures that the first character is a letter, and truncates the label to 15 characters. The Console displays the result, and if you don't like it, you can instead enter your own value in the **DNS Label** field. See Requirements for DNS Labels and Hostnames.

2. When creating the subnets:
    - Again, check the checkbox for **Use DNS Hostnames in this Subnet**
    - Specify a DNS label of your choice for each subnet. If you check the checkbox but don't specify the DNS label for a given subnet, the Console assumes you want to use the Internet and VCN Resolver for the subnet and automatically generates a DNS label for the subnet. The Console takes the subnet name you provided, removes non-alphanumeric characters, ensures that the first character is a letter, and truncates the label to 15 characters. The Console displays the result, and if you don't like it, you can instead enter your own value in the **DNS Label** field. See Requirements for DNS Labels and Hostnames.
    - Associate any set of DHCP options that has DNS type = Internet and VCN Resolver. The default set of DHCP options in the VCN uses the Internet and VCN Resolver by default.
3. When launching instances:
    - Specify a hostname (or at least a display name) for each instance. For more information, see Validation and Generation of the Hostname.

If you don't check the checkbox for **Use DNS Hostnames in this VCN** when creating the VCN, you can't set the DNS label for the VCN or subnets, and you can't specify a hostname during instance launch.



Note

The above procedure assumes you create the VCN and subnets one at a time in the Console. The Console has a feature that automatically creates a VCN with subnets and an internet gateway all at the same time. If you use that feature to create the VCN and subnets, the Console automatically generates DNS labels for them.

If you create a new VCN and subnets with the API

1. When creating the VCN:
    - Specify a DNS label for the VCN. See Requirements for DNS Labels and Hostnames. If you don't set a value (if it's null), Oracle assumes you don't want to use the Internet and VCN Resolver, even if the DHCP options have DhcpDnsOption `serverType` = `VcnLocalPlusInternet`.
2. When creating the subnets:
    - Specify a DNS label for each subnet. See Requirements for DNS Labels and Hostnames. If you specified a DNS label for the VCN, but you don't specify a DNS label for the subnet, Oracle assumes you don't want the instances in the subnet to use the Internet and VCN Resolver. They will not be able to use hostnames to communicate with instances in the VCN.
    - Associate any set of DHCP options that has DhcpDnsOption`serverType` = `VcnLocalPlusInternet`. The default set of DHCP options in the VCN uses this by default.
3. When launching instances:
    - Specify a hostname (or at least a display name) for each instance. For more information, see Validation and Generation of the Hostname.

If you don't specify a DNS label when creating the VCN, you can't set the DNS label for the subnets (the `CreateSubnet` call will fail), nor specify a hostname during instance launch (the `LaunchInstance` call will fail). You also cannot assign a hostname to a secondary VNIC or a secondary private IP.

Scenario 1: Use Internet and VCN Resolver with DNS Hostnames Across the VCN

The typical scenario is to enable the Internet and VCN Resolver *across your entire VCN*. This means all instances in the VCN can communicate with each other without knowing their IP addresses. To do that, follow the instructions for creating a new VCN in How to Enable DNS Hostnames in Your VCN, and make sure to assign a DNS label to the VCN and every subnet. Then make sure to assign every instance a hostname (or at least a display name) at launch. If you add a secondary VNIC or secondary private IP, also assign it a hostname. The instances can then communicate with each other using FQDNs instead of IP addresses. If you also set the Search Domain DHCP option to the VCN domain name (*<VCN DNS label>*.oraclevcn.com), the instances can then communicate with each other using just *<hostname>*.*<subnet DNS label>* instead of the FQDN.

Scenario 2: Use Custom DNS Servers to Resolve DNS Hostnames

You can set up an instance to be a custom DNS server within your VCN and configure it to resolve the hostnames that you set when launching the instances. You must configure the servers to use 169.254.169.254 as the forwarder for the VCN domain (that is, *<VCN DNS label>*.oraclevcn.com).



Note

The custom DNS servers must be located in a subnet that uses Internet and VCN Resolver for DNS.

For an example of an implementation of this scenario with the Oracle Terraform provider, see Hybrid DNS Configuration.

Scenario 3: Use Different DHCP Options Per Subnet

Scenario 1 assumes you want to use the Internet and VCN Resolver the same way across all subnets, and thus all instances in the VCN. You could, however, configure different DNS settings *for each subnet*, because the DHCP options are configured *at the subnet level*. The important thing to understand is this: the subnet where you want to generate the DNS query is where you need to configure the corresponding Internet and VCN Resolver settings.

For example, if you want instance A in subnet A to resolve the hostname of instance B in subnet B, you must configure subnet A to use the Internet and VCN Resolver. Conversely, if you want instance B to resolve the hostname of instance A, you must configure subnet B to use the Internet and VCN Resolver.

You can configure a different set of DHCP options for each subnet. For example, you could set subnet A's Search Domain to `subneta.vcn1.oraclevcn.com`, which means all instances in subnet A could use just hostnames to communicate with each other. You could similarly set subnet B's Search domain to `subnetb.vcn1.oraclevcn.com` to enable Subnet B's instances to communicate with each other with just hostnames. But that means any instances in a given subnet would need to use FQDNs to communicate with instances in other subnets.

DHCP Options

This topic describes how to manage the Dynamic Host Configuration Protocol (DHCP) options in a virtual cloud network (VCN).



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Overview of DHCP Options

The Networking service uses DHCP to automatically provide configuration information to instances when they boot up. Although DHCP lets you change some settings dynamically, others are static and never change. For example, when you launch an instance, either you or Oracle specifies the instance's private IP address. Each time the instance boots up or you restart the instance's DHCP client, DHCP passes that same private IP address to the instance. The address never changes during the instance's lifetime.

The Networking service provides *DHCP options* to let you control certain types of configuration on the instances in your VCN. You can change the values of these options at your discretion, unlike the static information that DHCP provides to the instance. The changes take effect the next time you restart a given instance's DHCP client or reboot the instance. For more details, see Important Notes about Your Instances and DHCP Options.

Each subnet in a VCN can have a single set of DHCP options associated with it. That set of options applies to all instances in the subnet. Each VCN comes with a *default set of DHCP options* with initial values that you can change. If you don't specify otherwise, every subnet uses the VCN's default set of DHCP options.

The following table summarizes the available DHCP options you can configure.

DHCP Option	Possible Values	Initial Value in the Default DHCP Options	Notes
Domain Name Server	DNS Type: • Internet and VCN Resolver • Custom Resolver	DNS Type = Internet and VCN resolver. For more information, see Choices for DNS in Your VCN.	If you set DNS Type = Custom Resolver, you can specify up to three DNS servers of your choice. For more information, see Choices for DNS in Your VCN.

DHCP Option	Possible Values	Initial Value in the Default DHCP Options	Notes
Search Domain	A single search domain	If you've set up your VCN with a DNS label, the default value for the Search Domain option is the VCN domain name (*<VCN DNS label>*.oraclevcn.com). Otherwise, the Search Domain option is not present in the default set of DHCP options.	In general, when *any* set of DHCP options is initially created (the default set or a custom set you create), the Networking service automatically adds the Search Domain option and sets it to the VCN domain name (*<VCN DNS label>*.oraclevcn.com) *if all of these are true*: • The VCN has a DNS label • DNS Type = Internet and VCN Resolver • You did NOT specify a search domain of your choice during creation of the set of DHCP options After the set of DHCP options is created, you can always remove the Search Domain option or set it to a different value. You can specify only a single search domain in a set of DHCP options.

Working with DHCP Options

When you create a subnet, you specify which set of DHCP options to associate with the subnet. If you don't, the default set of DHCP options for the VCN is used. You can change which set of

DHCP options the subnet uses at any time.

When creating a new set of DHCP options, you may optionally assign it a friendly name. It doesn't have to be unique, and you can change it later. Oracle automatically assigns the set of options a unique identifier called an Oracle Cloud ID (OCID). For more information, see Resource Identifiers.

You can change the values of an individual DHCP option in a set, but notice that when you use the REST API to update a single option in a set, the new set of options replaces the entire existing set.

To delete a set of DHCP options, it must not be associated with a subnet yet. You can't delete a VCN's default set of DHCP options.

For information about the maximum number of DHCP options allowed, see Service Limits.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: see IAM Policies for Networking.

Important Notes about Your Instances and DHCP Options

Whenever you change the value of one of the DHCP options, you must do one of the following for the change to take effect on existing instances in the subnets associated with that set of DHCP options: either restart the DHCP client on the instance, or reboot the instance.

Make sure to keep the DHCP client running so you can always access the instance. If you stop the DHCP client manually or disable NetworkManager (which stops the DHCP client on Linux instances), the instance can't renew its DHCP lease and will become inaccessible when the lease expires (typically within 24 hours). Do not disable NetworkManager unless you use another method to ensure renewal of the lease.

Stopping the DHCP client might remove the host route table when the lease expires. Also, loss of network connectivity to your iSCSI connections might result in loss of the boot drive.

Any changes you make to the `/etc/hosts` and `/etc/resolv.conf` files will be overwritten whenever the DHCP lease is renewed or the instance is rebooted. To persist your changes, add the following line to `/etc/oci-hostname.conf`:

```
PRESERVE_HOSTINFO=2
```

If the `/etc/oci-hostname.conf` file does not exist, create it.

Using the Console

To view a VCN's set of default DHCP options

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Under **Resources**, click **DHCP Options**.
   The default set and its details are displayed in the list.

To update options in an existing set of DHCP options

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Under **Resources**, click **DHCP Options**.
4. For the set you're interested in, click the Actions icon (three dots), and then click **Edit:**
   - For **DNS Type**: If want instances in the subnet to resolve internet hostnames and hostnames of instances in the VCN, select **Internet and VCN Resolver**. Or to use a DNS server of your choice, select **Custom Resolver** and then enter the server's IP address (three servers maximum). For more information, see DNS in

Your Virtual Cloud Network.

- For **Search Domain**: If you want instances in the subnet to append a particular search domain when resolving DNS queries, enter it here. If the Search Domain option is already set to the VCN domain name and you're not sure why, see the details in Overview of DHCP Options.

5. When you're done, click **Save Changes**.
6. If you have any existing instances in a subnet that uses this set of DHCP options, make sure to restart the DHCP client on each affected instance, or reboot the instance itself so that it picks up the new setting.

To create a new set of DHCP options

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Under **Resources**, click **DHCP Options**.
4. Click **Create DHCP Options**.
5. Enter the following:
   - **Name**: A friendly name for the set of options. It doesn't have to be unique, and you can change it later. Avoid entering confidential information.
   - **Create in Compartment:** The compartment where you want to create the set of DHCP options, if different from the compartment you're currently working in.
   - **DNS Type**: If want instances in the subnet to resolve internet hostnames and hostnames of instances in the VCN, select **Internet and VCN Resolver**. Or to use a DNS server of your choice, select **Custom Resolver** and then enter the server's IP address (three servers maximum). For more information, see DNS in Your Virtual Cloud Network.
   - **Search Domain**: If you want instances in the subnet to append a particular search domain when resolving DNS queries, enter it here. Be aware that the

Networking service automatically sets the Search Domain option in certain situations. See the details in Overview of DHCP Options.

- **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

6. When you're done, click **Create DHCP Options**.

The set of options is created and then displayed on the **DHCP Options** page of the compartment you chose. You can now specify this set of options when creating or updating a subnet.

To change which set of DHCP options a subnet uses

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Click **Subnets**.
4. Click the subnet you're interested in.
5. Click **Edit**.
6. In the **DHCP Options** section, select the new set of DHCP options you want the subnet to use.
7. Click **Save Changes**.
   The changes take effect within a few seconds.

To delete a set of DHCP options

Prerequisite: To delete a set of DHCP options, it must not be associated with a subnet yet. You

can't delete the default set of DHCP options in a VCN.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Under **Resources**, click **DHCP Options**.
4. For the set you want to delete, click the Actions icon (three dots), and then click **Terminate**.
5. Confirm when prompted.

To manage tags for a set of DHCP options

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Under **Resources**, click **DHCP Options**.
4. For the set you're interested in, click the Actions icon (three dots), and then click **View Tags**. From there you can view the existing tags, edit them, and apply new ones.

For more information, see Resource Tags.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

To manage a VCN's DHCP options, use these operations:

- ListDhcpOptions
- GetDhcpOptions
- UpdateDhcpOptions

- CreateDhcpOptions
- DeleteDhcpOptions

Route Tables

This topic describes how to manage the route tables in a virtual cloud network (VCN).



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Overview of Routing for Your VCN

Your VCN uses virtual route tables to send traffic out of the VCN (for example, to the internet, to your on-premises network, or to a peered VCN). These virtual route tables have rules that look and act like traditional network route rules you might already be familiar with. Each rule specifies a destination CIDR block and the target (the next hop) for any traffic that matches that CIDR.

Here are basics about routing in your VCN:

- The primary routing scenario is for sending a subnet's traffic to destinations outside the VCN. A subnet has a single route table of your choice associated with it. All VNICs in that subnet are subject to the rules in the route table. The rules govern how the traffic leaving the subnet is routed.
- When routing traffic, Oracle uses a subnet's route table *only* if the destination IP address is not within the VCN's CIDR block. No route rules are required in order to enable traffic within the VCN itself.

- If a route table has overlapping rules, Oracle uses the most specific rule in the table to route the traffic (that is, the rule with the longest prefix match).
- If there is no route rule that matches the network traffic you intend to route outside the VCN, the traffic is dropped (blackholed).

Working with Route Tables and Route Rules

Each VCN automatically comes with a default route table that has no rules. If you don't specify otherwise, every subnet uses the VCN's default route table. When you add route rules to your VCN, you can simply add them to the default table if that suits your needs. However, if you need both a public subnet and a private subnet (for example, see Scenario C: Public and Private Subnets with a VPN), you instead create a separate (custom) route table for each subnet.

Each subnet in a VCN uses a single route table. When you create the subnet, you specify which one to use. You can change which route table the subnet uses at any time. You can also edit a route table's rules, or remove all the rules from the table.

You may optionally assign a descriptive name to a custom route table during creation. It doesn't have to be unique, and you can change it later. Oracle automatically assigns the route table a unique identifier called an Oracle Cloud ID (OCID). For more information, see Resource Identifiers.

A route rule specifies a destination CIDR block and the target (the next hop) for any traffic that matches that CIDR. Here are the allowed types of targets for a route rule:

- dynamic routing gateway (DRG): For subnets that need private access to networks connected to your VCN (for example, your on-premises network connected with an IPSec VPN or FastConnect, or a peered VCN in another region).
- internet gateway: For public subnets that need direct access to the internet.
- NAT gateway: For subnets with instances that do not have public IP addresses but need outbound access to the internet.
- service gateway: For subnets that need private access to Oracle services such as Object Storage.

- local peering gateway (LPG): For subnets that need private access to a peered VCN in the same region.
- private IP: For subnets that need to route traffic to an instance in the VCN. For more information, see Using a Private IP as a Route Target.



Note

You can't delete a particular resource if it's the target in a route rule. For example, you can't delete an internet gateway that has traffic routed to it. You must first delete all rules (in all route tables) with that internet gateway as the target.

When adding a route rule to a route table, you provide the destination CIDR block and target (plus the compartment where the target resides). Exception: if the target is a service gateway, instead of a destination CIDR block, you specify an Oracle-provided string that represents the public endpoints for the service of interest. That way you don't need to know all the service's CIDR blocks, which might change over time.

If you misconfigure a rule (for example, enter the wrong destination CIDR block), the network traffic you intended to route might be dropped (blackholed) or sent to an unintended target.

You can't delete a VCN's default route table. To delete a custom route table, it must not be associated with a subnet yet. Or, in the advanced scenario of transit routing, it must not be associated with a DRG attachment or LPG in the VCN.

For information about the maximum number of route tables and route rules, see Service Limits.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have

permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: see IAM Policies for Networking.

Advanced Scenario: Transit Routing

There's an advanced routing scenario called transit routing that enables communication between an on-premises network and multiple VCNs over a *single* Oracle Cloud Infrastructure FastConnect or IPSec VPN. The scenario involves creating route tables that you associate *with a DRG attachment or an LPG on that VCN*, instead of a subnet. The Networking service imposes restrictions on how the route tables can be used:

- If you associate a route table with the DRG attachment on a VCN, the route table can contain only rules that use an LPG on the VCN as the target.
- Similarly, if you associate a route table with an LPG, the route table can contain only rules that use the VCN's attached DRG as the target.

A DRG attachment or LPG can exist without a route table associated with it. However, after you associate a route table with a DRG attachment or LPG, there must always be a route table associated with it. But, you can associate a *different* route table. You can also edit the table's rules, or delete some or all of the rules.

Using a Private IP as a Route Target

If you're not familiar with the definition of a private IP, see Private IP Addresses. In short: a private IP is an object that contains a private IP address and related properties and has its own OCID.

General Use Cases

As mentioned earlier, Oracle uses a given subnet's route table *only* for traffic with a destination IP address outside the VCN. Typically you set up one or more rules to route that traffic to a gateway on the VCN (for example, a DRG connected to your on-premises network, or an internet gateway connected to the internet). However, you might want to route that traffic (going to destinations outside the VCN) through an instance in the VCN first. In that

case, you can use a private IP in the VCN as the target instead of a gateway on the VCN. Here are a few reasons you might do this:

- To implement a virtual network function (such as a firewall or intrusion detection) that filters outgoing traffic from instances.
- To manage an overlay network on the VCN, which lets you run container orchestration workloads.
- To implement Network Address Translation (NAT) in the VCN. Note that Oracle instead recommends using a NAT gateway with your VCN. In general, NAT enables outbound internet access for instances that don't have direct internet connectivity.

To implement these use cases, there's more to do than simply route traffic to the instance. There's also configuration required on the instance itself.



Tip

You can enable high availability of the private IP route target by using a secondary private IP address. In the event of a failure, you can move the secondary private IP from an existing VNIC to another VNIC in the same subnet. See To move a secondary private IP to another VNIC in the same subnet (Console instructions) and UpdatePrivateIp (API instructions).

Requirements for Using a Private IP as a Target

- The private IP must be in the same VCN as the route table.
- The private IP's VNIC must be configured to skip the source/destination check so that the VNIC can forward traffic. By default, VNICs are configured to perform the check. For more information, see Source/Destination Check.
- You must configure the instance itself to forward packets. For more information, see NAT Instance Configuration.

- The route rule must specify the OCID of the private IP as the target, and not the IP address itself. Exception: If you use the Console, you can instead specify the private IP address itself as the target, and the Console determines and uses the corresponding OCID in the rule.



Important

A route rule with a private IP target can result in blackholing in these cases:

- The instance the private IP is assigned to is stopped or terminated
- The VNIC the private IP is assigned to is updated to enable the source/destination check or is deleted
- The private IP is unassigned from the VNIC

When a target private IP is terminated, in the Console, the route rule displays a note that the target OCID no longer exists.

For failover: If your target instance is terminated before you can move the secondary private IP to a standby, you must update the route rule to use the OCID of the new target private IP on the standby. The rule uses the target's OCID and not the private IP address itself.

General Setup Process

1. Determine which instance will receive and forward the traffic (the NAT instance, for example).

2. Choose a private IP on the instance (can be on the instance's primary VNIC or a secondary VNIC). If you want to implement failover, set up a secondary private IP on one of the VNICs on the instance.
3. Disable the source/destination check on the private IP's VNIC. See Source/Destination Check.
4. Get the OCID for the private IP. If you're using the Console, you can get either the OCID or the private IP address itself, along with the name of the private IP's compartment.
5. For the subnet that needs to route traffic to the private IP, view the subnet's route table. If the table already has a rule with the same destination CIDR but a different target, delete that rule.
6. Add a route rule with the following:
    - **Destination CIDR block**: If all traffic leaving the subnet needs to go to the private IP, use 0.0.0.0/0.
    - **Target type:** Private IP.
    - **Compartment:** The compartment of the private IP.
    - **Target:** The OCID of the private IP. If you're using the Console and instead enter the private IP address itself, the Console determines the corresponding OCID and uses it as the target for the route rule.

As mentioned earlier, you must configure the instance itself to forward packets. For more information, see NAT Instance Configuration.

Using the Console

To view a VCN's default route table

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Under **Resources**, click **Route Tables.**

The default route table is displayed in the list of tables.

4. Click the default route table to view its details.

To update rules in an existing route table

You can add, edit, or delete rules.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Under **Resources**, click **Route Tables**.
4. Click the route table you're interested in.
5. If you want to create a new route rule, click **Add Route Rule** and enter the following:
   - **Target Type:** See the list of target types in Route Tables. If the target type is a DRG, the VCN's attached DRG is automatically selected as the target, and you don't have to specify the target yourself. If the target is a private IP, make sure you've first disabled the source/destination check on the private IP's VNIC. For more information, see Using a Private IP as a Route Target.
   - **Destination CIDR Block**: Only if the target is not a service gateway. The value is the destination CIDR block for the traffic. A value of 0.0.0.0/0 means that all non-intra-VCN traffic that is not already covered by other rules in the route table will go to the target specified in this rule.
   - **Destination Service:** Only if the target is a service gateway. The value is the service CIDR label that you're interested in.
   - **Compartment:** The compartment where the target is located.
   - **Target:** The target. If the target is a private IP, enter its OCID. Or you can enter the private IP address itself, in which case the Console determines the corresponding OCID and uses it as the target for the route rule.
6. If you want to delete an existing rule, click the Actions icon (three dots), and then click **Remove**.

7. If you wanted to edit an existing rule, click the Actions icon (three dots), and then click **Edit**.

To create a route table

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Under **Resources**, click **Route Tables**.
4. Click **Create Route Table**.
5. Enter the following:
   - **Name:** A friendly name for the route table. The name doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
   - **Create in Compartment:** The compartment where you want to create the route table, if different from the compartment you're currently working in.
6. Optionally, click **+Additional Route Rule** to add one or more route rules, each with the following information (remember, a route table can exist with no rules until you're ready to add them):
   - **Target Type:** See the list of target types in Route Tables. If the target type is a DRG, the VCN's attached DRG is automatically selected as the target, and you don't have to specify the target yourself. If the target is a private IP, make sure you've first disabled the source/destination check on the private IP's VNIC. For more information, see Using a Private IP as a Route Target.
   - **Destination CIDR Block**: Only if the target is not a service gateway. The value is the destination CIDR block for the traffic. A value of 0.0.0.0/0 means that all non-intra-VCN traffic that is not already covered by other rules in the route table will go to the target specified in this rule.

- **Destination Service:** Only if the target is a service gateway. The value is the service CIDR label that you're interested in.
- **Compartment:** The compartment where the target is located.
- **Target:** The target. If the target is a private IP, enter its OCID. Or you can enter the private IP address itself, in which case the Console determines the corresponding OCID and uses it as the target for the route rule.

7. **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
8. Click **Create Route Table**.
   The route table is created and then displayed on the **Route Tables** page in the compartment you chose. You can now specify this route table when creating or updating a subnet.

To change which route table a subnet uses

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Click **Subnets**.
4. Click the subnet you're interested in
5. Click **Edit**.
6. In the **Route Table** section, select the new route table you want the subnet to use.
7. Click **Save Changes**.
   The changes take effect within a few seconds.

To delete a route table

Prerequisite: To delete a route table, it must not be associated with a subnet yet. You can't delete the default route table in a VCN.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Under **Resources**, click **Route Tables**.
4. Click the route table you're interested in.
5. Click **Terminate**.
6. Confirm when prompted.

To manage tags for a route table

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Under **Resources**, click **Route Tables**.
4. Click the route table you're interested in.
5. Click the **Tags** tab to view or edit the existing tags. Or click **Apply tag(s)** to add new ones.

For more information, see Resource Tags.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

To manage a VCN's route tables, use these operations:

- ListRouteTables
- GetRouteTable
- UpdateRouteTable
- CreateRouteTable
- DeleteRouteTable

Dynamic Routing Gateways (DRGs)

This topic describes how to manage a dynamic routing gateway (DRG). This topic uses the terms *dynamic routing gateway* and *DRG* interchangeably. The Console uses the term *Dynamic Routing Gateway*, whereas for brevity the API uses *DRG*.

You use a DRG when connecting your existing on-premises network to your virtual cloud network (VCN) with one (or both) of these:

- IPSec VPN
- Oracle Cloud Infrastructure FastConnect

You also use a DRG when peering a VCN with a VCN in a different region:

- Remote VCN Peering (Across Regions)



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Overview of Dynamic Routing Gateways

You can think of a DRG as a virtual router that provides a path for private traffic (that is, traffic that uses private IPv4 addresses) between your VCN and networks outside the VCN's region.

For example, if you use an IPSec VPN or Oracle Cloud Infrastructure FastConnect (or both) to connect your on-premises network to your VCN, that private IPv4 address traffic goes through a DRG that you create and attach to your VCN. For scenarios for using a DRG to connect a VCN to your on-premises network, see Overview of Networking.

Also, if you decide to peer your VCN with a VCN in another region, your VCN's DRG routes traffic to the other VCN over a private backbone that connects the regions (without traffic traversing the internet). For information about connecting VCNs in different regions, see Remote VCN Peering (Across Regions).

Working with DRGs and DRG Attachments

For the purposes of access control, when creating a DRG, you must specify the compartment where you want the DRG to reside. If you're not sure which compartment to use, put the DRG in the same compartment as the VCN. For more information, see Access Control.

You may optionally assign a friendly name to the DRG. It doesn't have to be unique, and you can change it later. Oracle automatically assigns the DRG a unique identifier called an Oracle Cloud ID (OCID). For more information, see Resource Identifiers.

A DRG is a standalone object. To use it, you must attach it to a VCN. A VCN can be attached to only one DRG at a time, and a DRG can be attached to only one VCN at a time. You can detach a DRG and reattach it at any time. In the API, the process of attaching creates a `DrgAttachment` object with its own OCID. To detach the DRG, you delete that attachment object.

After attaching a DRG, you must update the routing in the VCN to use the DRG. Otherwise, traffic from the VCN will not flow to the DRG. See To route a subnet's traffic to a dynamic routing gateway.

To delete a DRG, it must not be attached to a VCN or connected to another network by way of IPSec VPN, Oracle Cloud Infrastructure FastConnect, or remote VCN peering. Also, there must not be a route rule that lists that DRG as a target.

For information about the number of DRGs you can have, see Service Limits.

Routing a Subnet's Traffic to a DRG

The basic routing scenario sends traffic from a subnet in the VCN to the DRG. For example, if you're sending traffic from the subnet to your on-premises network, you set up a rule in the subnet's route table. The rule's destination CIDR is the CIDR for the on-premises network (or a subnet within), and the rule's target is the DRG. For more information, see Route Tables.

Advanced Scenario: Transit Routing

There's an advanced routing scenario called transit routing that enables communication between an on-premises network and multiple VCNs over a *single* Oracle Cloud Infrastructure FastConnect or IPSec VPN. The VCNs must be in the same region and locally peered in a hub-and-spoke layout. The VCN that is acting as the hub is connected to the on-premises network by way of an attached DRG. That VCN has a route table *associated with its DRG attachment* (typically route tables are associated with a VCN's subnets). That route table lets you manage routing of traffic to the spoke VCNs.

When you attach a DRG to a VCN, you can optionally associate a route table with the attachment. Or if you already have a DRG attachment, you can associate a route table with it. The route table must belong to the attached VCN. A route table associated with a DRG attachment can contain only rules that use a local peering gateway (LPG) on the attached VCN as the target.

A DRG attachment can exist without a route table associated with it. However, after you associate a route table with a DRG attachment, there must always be a route table associated with it. But, you can associate a *different* route table. You can also edit the table's rules, or delete some or all of the rules.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: see IAM Policies for Networking.

Using the Console

In general, to use a DRG, you must complete these steps:

1. Create the DRG.
2. Attach the DRG to your VCN.
3. Route subnet traffic to the DRG. This involves updating the route table associated with each subnet that must send traffic to the DRG. If all the subnets use the VCN's default route table, you must only update that one table.

To create a dynamic routing gateway

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Dynamic Routing Gateways**.
2. Choose a compartment you have permission to work in (on the left side of the page). The page updates to display only the resources in that compartment. If you're not sure which compartment to use, contact an administrator. For more information, see Access Control.
3. Click **Create Dynamic Routing Gateway**.
4. Enter the following items:
    - **Create in Compartment:** The compartment where you want to create the DRG, if different from the compartment you're currently working in.

- **Name:** A friendly name for the DRG. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
- **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

5. Click **Create Dynamic Routing Gateway**.

The resource is created and then displayed on the **Dynamic Routing Gateways** page of the compartment you chose. It will be in the "Provisioning" state for a short period. You can connect it to other parts of your network only after provisioning is complete.

To attach a dynamic routing gateway to a VCN

Note: A VCN can be attached to only one DRG at a time, and a DRG can be attached to only one VCN at a time. The attachment is automatically created in the compartment that holds the VCN.

The following instructions have you navigate to the DRG and then choose which VCN to attach. You could instead navigate to the VCN and then choose which DRG to attach.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Dynamic Routing Gateways**.
2. Click the DRG you want to attach.
3. Under **Resources**, click **Virtual Cloud Networks**. If you want to attach the DRG to a VCN in a different compartment than the one you're working in, choose that compartment from the list on the left side of the page.
4. Click **Attach to Virtual Cloud Network**.
5. Enter the following items:

- **Virtual Cloud Network Compartment:** The compartment where the VCN resides.
- **Virtual Cloud Network:** The VCN.
- **Associate with Route Table (optional):** Only if you're setting up the advanced routing scenario called transit routing. Select the compartment that contains the route table you want to associate with the DRG attachment, and the route table itself. You can skip this part and associate the DRG attachment with a route table later if you like.

6. Click **Attach**.

The attachment will be in the "Attaching" state for a short period before it's ready.

After it's ready, make sure to create a route rule that directs subnet traffic to this DRG. See To route a subnet's traffic to a dynamic routing gateway.

To route a subnet's traffic to a dynamic routing gateway

For each subnet that must send traffic to the DRG, you must add a route rule to the route table associated with that subnet. If all the subnets in the VCN use the default route table, you must add a rule to only that one table.

If all non-intra-VCN traffic that's not covered by another rule in the table must be routed to the DRG, then this is the new rule to add:

- **Target Type:** Dynamic Routing Gateway. The VCN's attached DRG is automatically selected as the target, and you don't have to specify the target yourself.
- **Destination CIDR Block** = 0.0.0.0/0. If you want to limit the rule to a specific network (for example, your on-premises network), then use that network's CIDR instead of 0.0.0.0/0.

For step-by-step instructions, see To update rules in an existing route table.

To associate a route table with an existing dynamic routing gateway attachment

This task is related to an advanced routing scenario called transit routing.

A DRG attachment can exist without a route table associated with it. However, after you associate a route table with a DRG attachment, there must always be a route table associated with it. But, you can associate a *different* route table. You can also edit the table's rules, or delete some or all of the rules.

Prerequisites: The route table must exist and belong to the VCN that the DRG is already attached to.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Dynamic Routing Gateways**.
2. Click the DRG attached to the VCN with the route table.
3. Click the Actions icon (three dots), and then click either:
   - **Associate With Route Table:** If the DRG attachment has no route table associated with it yet.
   - **Associate Different Route Table:** If you're changing which route table is associated with the DRG attachment.
4. Select the compartment where the route table resides, and select the route table itself.
5. Click **Associate**.

The route table is associated with the DRG attachment.

To detach a dynamic routing gateway from a VCN

Note: You do not need to remove the route rule that routes traffic to the DRG before you detach the DRG from the VCN.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Dynamic Routing Gateways**.

2. Click the DRG you want to detach.
3. Under **Resources**, click **Virtual Cloud Networks** to see the VCN the DRG is attached to. If the VCN is in a different compartment than the one you're working in, choose that compartment from the list on the left side of the page.
4. Click the Actions icon (three dots), and then click **Detach**.

The attachment will be in the "Detaching" state for a short period.

To delete a dynamic routing gateway

Prerequisites: The DRG must not be attached to a cloud network. It must also not be connected to another network with an IPSec VPN, with Oracle Cloud Infrastructure FastConnect, or through remote VCN peering. Finally, there must not be a route rule that lists the DRG as a target.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Dynamic Routing Gateways**.
2. Click the DRG you're interested in.
3. Click **Terminate**.
4. Confirm when prompted.

The DRG will be in the "Terminating" state for a short period while it's being deleted.

To manage tags for a dynamic routing gateway

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Dynamic Routing Gateways**.
2. Click the DRG you're interested in.
3. Click the **Tags** tab to view or edit the existing tags. Or click **Apply tag(s)** to add new ones.

For more information, see Resource Tags.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

To manage your DRGs, use these operations:

- ListDrgs
- CreateDrg
- GetDrg
- UpdateDrg
- DeleteDrg
- ListDrgAttachments
- CreateDrgAttachment: This attaches a DRG to a VCN and results in a `DrgAttachment` object with its own OCID. You may optionally specify a route table if you're setting up the advanced routing scenario called transit routing.
- GetDrgAttachment
- UpdateDrgAttachment: Among other things, this associates a route table with an existing DRG attachment for transit routing.
- DeleteDrgAttachment: This detaches a DRG from a VCN by deleting the `DrgAttachment`.

For information about route table operations, see Route Tables.

VPN Connect

The following topics have information about setting up VPN Connect (also referred to as an IPSec VPN) between your on-premises network and virtual cloud network (VCN):

- VPN Connect Overview
- Supported IPSec Parameters
- Setting Up VPN Connect

- Configuring Your CPE
  - Generic CPE Configuration Information
  - Check Point
  - Cisco ASA:
    - Cisco ASA Configuration Options
    - Cisco ASA: Route-Based
    - Cisco ASA: Policy-Based Without VPN Filters
    - Cisco ASA: Policy-Based with VPN Filters
  - Cisco IOS
  - FortiGate
  - Juniper MX
  - Juniper SRX
  - Libreswan
  - Openswan
  - Palo Alto
  - WatchGuard
  - Yamaha RTX Series
- Verified CPE Devices
- Working with VPN Connect
- VPN Connect FAQ
- Using the API for VPN Connect
- VPN Connect Troubleshooting

VPN Connect Overview

One way to connect your on-premises network and your virtual cloud network (VCN) is to use VPN Connect, which is an IPSec VPN. IPSec stands for *Internet Protocol Security* or *IP*

Security. IPSec is a protocol suite that encrypts the entire IP traffic before the packets are transferred from the source to the destination.

This topic gives an overview of an IPSec VPN for your VCN. For scenarios that include an IPSec VPN, see Scenario B: Private Subnet with a VPN and Scenario C: Public and Private Subnets with a VPN.

Required Personnel and Knowledge

Typically the following types of personnel are involved in setting up an IPSec VPN with Oracle Cloud Infrastructure:

- **Dev Ops team member** (or similar function) who uses the Oracle Cloud InfrastructureConsole to set up the cloud components required for the virtual network and IPSec VPN.
- **Network engineer** (or similar function) who configures the customer-premises equipment (CPE) device with information provided by the Dev Ops team member.



Tip

The Dev Ops team member must have the required permission to create and manage the cloud components. If the person is the default administrator for your Oracle Cloud Infrastructure tenancy or a member of the Administrators group, then they have the required permission. For information about restricting access to your networking components, see Access Control.

The personnel should be familiar with the following concepts and definitions:

- The fundamentals of Oracle Cloud Infrastructure
- The basic Networking service components

- General IPSec VPN tunnel functionality

CLOUD RESOURCES

Anything you provision on a cloud platform. For example, with Oracle Cloud Infrastructure, a cloud resource can refer to a VCN, compute instance, user, compartment, load balancer, or any other service component on the platform.

ON-PREMISES

A widely used term in cloud technologies that refers to your traditional data center environments. On-premises can refer to a colocation scenario, a dedicated floor space, a dedicated data center building, or a desktop running under your desk.

ORACLE CLOUD IDENTIFIER (OCID)

A unique identifier assigned to each resource that you provision on Oracle Cloud Infrastructure. The OCID is a long string that Oracle automatically generates. You can't choose the value for an OCID or change a resource's OCID. For more information, see Resource Identifiers.

About the Oracle IPSec VPN

In general, IPSec can be configured in the following modes:

- **Transport mode:** IPSec encrypts and authenticates only the actual payload of the packet, and the header information stays intact.
- **Tunnel mode (supported by Oracle):** IPSec encrypts and authenticates the entire packet. After encryption, the packet is then encapsulated to form a new IP packet that has different header information.

Oracle Cloud Infrastructure supports only the tunnel mode for IPSec VPNs.

Each Oracle IPSec VPN consists of multiple redundant IPSec tunnels. For a given tunnel, you can use either Border Gateway Protocol (BGP) dynamic routing or static routing to route that tunnel's traffic. More details about routing follow.

IPSec VPN site-to-site tunnels offer the following advantages:

- Public internet lines are used to transmit data, so dedicated, expensive lease lines from one site to another aren't necessary.
- The internal IP addresses of the participating networks and nodes are hidden from external users.
- The entire communication between the source and destination sites is encrypted, significantly lowering the chances of information theft.

Routing for the Oracle IPSec VPN

When you create an IPSec VPN, it has two redundant IPSec tunnels. Oracle encourages you to configure your CPE device to use both tunnels (if your device supports it). Note that in the past, Oracle created IPSec VPNs that had up to four IPSec tunnels.

The following two routing types are available, and you choose the routing type separately *for each tunnel* in the IPSec VPN:

- **BGP dynamic routing:** The available routes are learned dynamically through BGP. The DRG dynamically learns the routes from your on-premises network. On the Oracle side, the DRG advertises the VCN's subnets.
- **Static routing:** When you set up the IPSec connection to the DRG, you specify the particular routes to your on-premises network that you want the VCN to know about. You also must configure your CPE device with static routes to the VCN's subnets. None of these routes are learned dynamically.

Important Routing Details for an Oracle IPSec VPN

Here are important details to understand about routing for your IPSec VPN:

- **Routing choices:**
    - Originally, the Oracle IPSec VPN supported only static routing, and you were required to provide at least one static route for the overall IPSec connection.

- Now two different types of routing are available (BGP and static routing), and you configure the routing type *per tunnel*. *Only one type of routing at a time is supported for a given tunnel*.
- In general, Oracle encourages you to use the same routing type for all tunnels in your IPSec connection. Exception: if you're in the process of transitioning between static routing and BGP, then one tunnel might temporarily still use static routing while the other has already been switched to BGP.
- When you create an IPSec connection, static routing is the default type of routing for all tunnels *unless you explicitly configure each tunnel to use BGP*.

- **Routing information required:**
  - If you choose BGP, for each tunnel you must provide two IP addresses (one for each of the two BGP speakers in the tunnel's BGP session). The addresses must be in the encryption domain for the IPSec connection. You must also provide the BGP autonomous system number (BGP ASN) for your network.
  - If you choose static routing, you must provide at least one static route (maximum 10). The static routes are configured with the *overall IPSec connection*, so the same set of static routes are used for *all* tunnels in the IPSec connection that are configured to use static routing. You can change the static routes at any time after creating the IPSec connection. If you're doing PAT between your CPE device and VCN, the static route for the IPSec connection is the PAT IP address. See Example Layout with PAT.
  - If you choose static routing, you may optionally provide an IP address for each end of the tunnel for the purposes of tunnel troubleshooting or monitoring.
  - If the tunnel is configured to use BGP, the IPSec connection's static routes are ignored. Any static routes associated with the IPSec connection are used for routing a given tunnel's traffic *only if that tunnel is configured to use static routing*. This is especially relevant if you have an IPSec VPN that uses static routing, but want to switch to using BGP.
- **Changing the routing:**

- If you want to change a tunnel from BGP to static routing, you must first ensure that the IPSec connection itself has at least one static route associated with it.
- You can change an existing tunnel's routing type at any time (unless the tunnel is currently being provisioned by Oracle). While you change the routing, the tunnel remains up (its IPSec status does not change). However, traffic flowing through the tunnel is disrupted temporarily during reprovisioning and while you reconfigure your CPE device. For information about making changes to an existing IPSec VPN, see Working with VPN Connect.
- Because you configure the routing type separately for each tunnel, if you want to switch your IPSec VPN from static routing to BGP, you can do it one tunnel at a time. This avoids the entire IPSec VPN being down. For instructions, see Changing from Static Routing to BGP Dynamic Routing.

Routing Preferences When You Have Redundant Connections

For redundancy, you might use multiple connections between your on-premises network and VCN. For example, you might use both a FastConnect and an IPSec VPN. Or perhaps you use multiple IPSec VPNs (for an example scenario, see Example Layout with Multiple Geographic Areas). The IPSec VPNs could use either BGP or static routing, or a combination. FastConnect always uses BGP.

When responding to a request or initiating new connections, **Oracle uses the shortest AS path. This means asymmetric routing is allowed.** Therefore Oracle's response to a request can follow a different path than the request. For example, depending on how routing is configured, you could send a request over IPSec VPN, but the Oracle response could come back over FastConnect. If you want to force routing to be symmetric, Oracle recommends using BGP and AS path prepending with your routes to influence which path Oracle uses when responding to and initiating connections.

Oracle implements AS path prepending to determine which path to use if the same route is advertised over multiple connections between your on-premises network and VCN. The details are summarized in the following table. Assuming that you're not influencing routing in some way, when the same route is advertised over multiple paths to the dynamic routing gateway (DRG) at the Oracle end of the connections, Oracle prefers the paths in the following order:

Oracle preference	Path	Details of how Oracle prefers the path	Resulting AS path for the route
1	FastConnect	Oracle prepends no ASNs to the routes that your CPE device advertises.	Your ASN
2	IPSec VPN with BGP routing	Oracle prepends a single private ASN on all the routes that your CPE device advertises over IPSec VPN with BGP.	Private ASN, Your ASN
3	IPSec VPN with static routing	Oracle prepends 3 private ASNs on the static routes that you've provided (Oracle advertises those routes to the dynamic routing gateway (DRG) at the Oracle end of the IPSec VPN) .	Private ASN, Private ASN, Private ASN, Your ASN

If you have two connections of the same type (for example, two IPSec VPNs that both use BGP), and you advertise the *same* routes across both connections, Oracle prefers the oldest established route when responding to requests or initiating connections.

Preferring a Specific Tunnel in the IPSec VPN

Within an IPSec VPN, you can influence *which tunnel* is preferred. Here are items you can configure:

- **Your CPE's BGP local preference:** If you use BGP, you can configure the BGP local preference attribute on your CPE device to control which tunnel is preferred for connections initiated from your on-premises network to your VCN. Because Oracle generally uses asymmetric routing, you must configure other attributes if you want Oracle to respond on that same tunnel. See the next two items.

- **More specific routes on the preferred tunnel:** You can configure your CPE to advertise more specific routes for the tunnel that you want to prefer. Oracle uses the route with the longest prefix match when responding to or initiating connections.
- **AS path prepending:** BGP prefers the shortest AS path, so if you use BGP, you can use AS path prepending to control which tunnel has the shortest path for a given route. Oracle uses the shortest AS path when responding to or initiating connections.

Routes Advertised to Your On-Premises Network

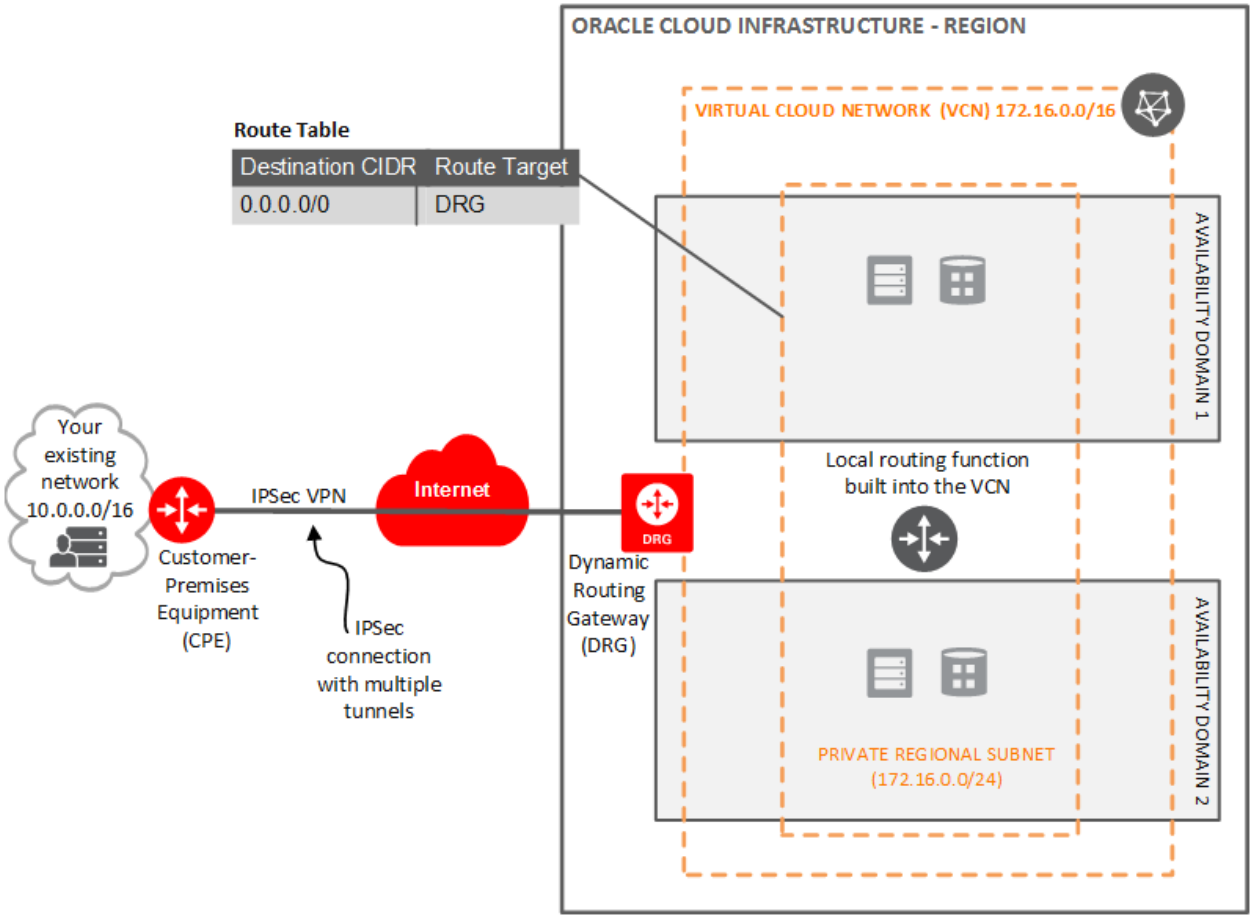
The dynamic routing gateway at the Oracle end of a FastConnect or IPSec VPN advertises the individual subnets in the DRG's attached VCN. A DRG can be attached to only a single VCN, and a VCN can be attached to only a single DRG.

However, if you set up transit routing, the DRG advertises additional routes. Transit routing is an advanced routing scenario that involves a single FastConnect or IPSec VPN and multiple *peered* VCNs in a hub-and-spoke layout. With transit routing, the DRG advertises additional routes for the VCNs that are peered with the DRG's attached VCN (the hub). For more information, see Advanced Scenario: Transit Routing.

Overview of the IPSec VPN Components

If you're not already familiar with the basic Networking service components, see Overview of Networking before proceeding.

When you set up an IPSec VPN for your VCN, you must create several Networking components. You can create the components with either the Console or the API. See the following diagram and description of the components.



CPE OBJECT

At your end of the IPSec VPN is the actual device in your on-premises network (whether hardware or software). The term *customer-premises equipment (CPE)* is commonly used in some industries to refer to this type of on-premises equipment. When setting up the

VPN, you must create a *virtual representation* of the device. Oracle calls the virtual representation a CPE, but this documentation typically uses the term *CPE object* to help distinguish the virtual representation from the actual CPE device. The CPE object contains basic information about your device that Oracle needs.

DYNAMIC ROUTING GATEWAY (DRG)

At Oracle's end of the IPSec VPN is a virtual router called a dynamic routing gateway, which is the gateway into your VCN from your on-premises network. Whether you're using an IPSec VPN or Oracle Cloud Infrastructure FastConnect private virtual circuits to connect your on-premises network and VCN, the traffic goes through the DRG. For more information, see Dynamic Routing Gateways (DRGs).

A network engineer might think of the DRG as the *VPN headend*. After creating a DRG, you must *attach* it to your VCN, using either the Console or API. You must also add one or more route rules that route traffic from the VCN to the DRG. Without that DRG attachment and the route rules, traffic does not flow between your VCN and on-premises network. At any time, you can detach the DRG from your VCN but maintain all the remaining VPN components. You can then reattach the DRG, or attach it to another VCN.

IPSEC CONNECTION

After creating the CPE object and DRG, you connect them by creating an IPSec connection, which you can think of as a parent object that represents the overall IPSec VPN. The IPSec connection has its own OCID. When you create this component, you configure the type of routing to use for each tunnel, and you provide any related routing information.

TUNNEL

An IPSec tunnel is used to encrypt traffic between secure IPSec endpoints. Oracle creates two tunnels in each IPSec connection for redundancy. Each tunnel has its own OCID. Oracle recommends that you configure your CPE device to support both tunnels in case one fails or Oracle takes one offline for maintenance. Each tunnel has configuration information that your network engineer needs when configuring your CPE device. This information includes an IP address and shared secret, as well as ISAKMP and IPSec parameters. For more information, see Supported IPSec Parameters and Verified CPE Devices.

Access Control for the Components

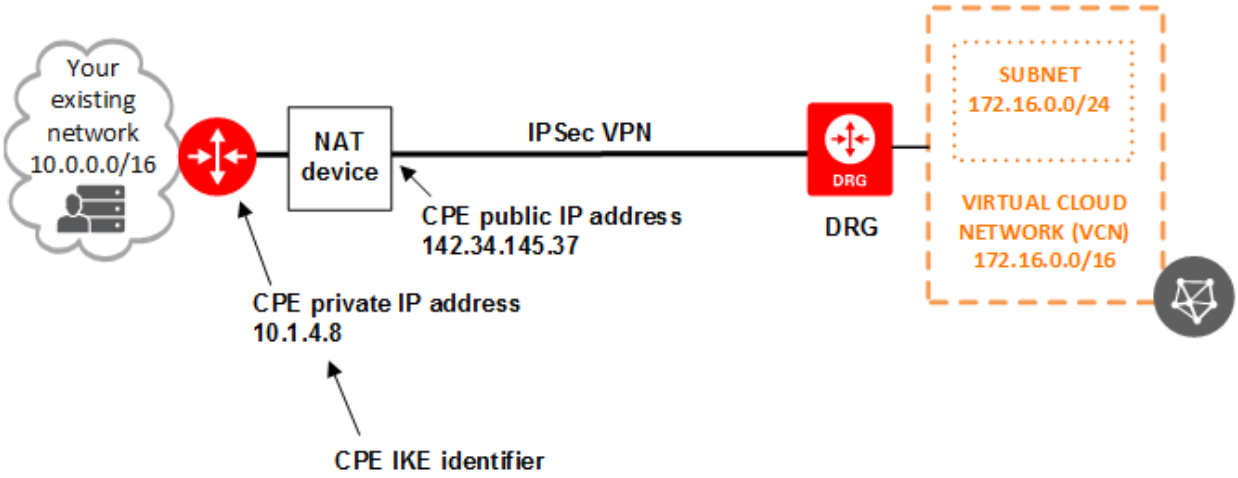
For the purposes of access control, when you set up the IPSec VPN, you must specify the compartment where you want each of the components to reside. If you're not sure which compartment to use, put all the components in the same compartment as the VCN. Note that the IPSec tunnels always reside in the same compartment as the parent IPSec connection. For information about compartments and restricting access to your networking components, see Access Control.

Component Names and Identifiers

You can optionally assign a descriptive name to each of the components when you create them. These names don't have to be unique, although it's a best practice to use unique names across your tenancy. Avoid including confidential information in the names. Oracle automatically assigns each component an OCID. For more information, see Resource Identifiers.

If Your CPE Is Behind a NAT Device

In general, the CPE IKE identifier configured on your end of the connection must match the CPE IKE identifier that Oracle is using. By default, Oracle uses the CPE's *public* IP address, which you provide when you create the CPE object. However, if your CPE is behind a NAT device, the CPE IKE identifier configured on your end might be the CPE's *private* IP address, as show in the following diagram.



If you cannot change your CPE's IKE identifier (some CPE platforms do not allow you to), you must provide Oracle the CPE IKE identifier value you're using on your end. You can provide the value either when you set up the IPSec connection, or later, by editing the IPSec connection. Oracle expects the value to be either an IP address or a fully qualified domain name (FQDN) such as *cpe.example.com*.

For instructions, see Changing the CPE IKE Identifier That Oracle Uses.

About the Tunnel Shared Secret

Each tunnel has a shared secret. By default, Oracle assigns the shared secret to the tunnel unless you provide a shared secret yourself. You can provide a shared secret for each tunnel when you create the IPSec connection, or later after the tunnels are created. For the shared secret, only letters, numbers, and spaces are allowed. If you change an existing tunnel's shared secret, the tunnel goes down while it is being reprovisioned.

For instructions, see Changing the Shared Secret That an IPSec Tunnel Uses

What's Next?

See these related topics:

- Setting Up VPN Connect
- Supported IPSec Parameters
- Configuring Your CPE
- Verified CPE Devices
- Working with VPN Connect
- VPN Connect FAQ
- Using the API for VPN Connect

Supported IPSec Parameters

This topic lists the supported ISAKMP and IPSec configuration parameters for an IPSec VPN. Oracle chose these values to maximize security and to cover a wide range of CPE devices. For some parameters, Oracle supports multiple values, and the recommended one is highlighted in *red italics*. If your CPE device is not on the list of verified devices, use the information here to configure your device.



Important

Oracle uses asymmetric routing across the multiple tunnels that make up the IPSec VPN connection. Even if you configure one tunnel as primary and another as backup, traffic from your VCN to your on-premises network can use any tunnel that is "up" on your device. Configure your firewalls accordingly. Otherwise, ping tests or application traffic across the connection will not reliably work.

ISAKMP Policy Options

- ISAKMP Protocol version 1
- Exchange type: Main mode
- Authentication method: pre-shared-keys
- Encryption: *AES-256-cbc*, AES-192-cbc, AES-128-cbc
- Authentication algorithm: *SHA-384*, SHA-256, SHA1 (also called SHA or SHA1-96)
- Diffie-Hellman group: *group 5*, group 2, group 1
- IKE session key lifetime: 28800 seconds (8 hours)

IPSec Policy Options

- IPSec protocol: ESP, tunnel-mode
- Encryption: *AES-256-cbc*, AES-192-cbc, AES-128-cbc
- Authentication algorithm: HMAC-SHA1-96
- IPSec session key lifetime: 3600 seconds (1 hour)
- Perfect Forward Secrecy (PFS): enabled, group 5

Security Parameter Index

The values for the Security Parameter Index (SPI) depend on whether your CPE supports route-based tunnels or policy-based tunnels. For more information about the correct SPI values to use, see Route-Based Versus Policy-Based IPSec.

Setting Up VPN Connect

This topic gives instructions for setting up VPN Connect for your VCN. For general information about IPSec VPNs, see VPN Connect Overview.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Before You Get Started

To prepare, do these things first:

- Read this section: Routing for the Oracle IPSec VPN
- Answer these questions:

Question	Answer
What is your VCN's CIDR?	
What is the public IP address of your CPE device? If you have multiple devices for redundancy, get the IP address for each. **Note:** If your CPE device is behind a NAT device, see If Your CPE Is Behind a NAT Device and also Information About Your CPE Device.	
Will you be doing port address translation (PAT) between each CPE device and your VCN?	

Question	Answer
What type of routing do you plan to use? If it's BGP dynamic routing, what are the BGP session IP addresses to use and the ASN of your network? The IP addresses must be part of the IPSec VPN's encryption domain. If static routing, what are the static routes for your on-premises network? See Routing for the Oracle IPSec VPN.	
Do you want to provide each tunnel's shared secret or let Oracle assign them? See About the Tunnel Shared Secret.	

- Draw a diagram of your network layout similar to the one in the preceding section. Think about which parts of your on-premises network need to communicate with your VCN, and the reverse. Map out the routing and security lists that you need for your VCN.



Tip

If you have an existing Oracle IPSec VPN that uses static routing, you can change the tunnels to instead use BGP dynamic routing.

Overall Process

Here's the overall process for setting up an IPSec VPN:

1. **Complete the tasks listed in** Before You Get Started**.**
2. **Set up the IPSec VPN components** (instructions in Example: Setting Up a Proof of Concept IPSec VPN):
    a. Create your VCN.
    b. Create a DRG.
    c. Attach the DRG to your VCN.

   d. Create a route table and route rule for the DRG.
   e. Create a security list and required rules.
   f. Create a subnet in the VCN.
   g. Create a CPE object and provide your CPE device's public IP address.
   h. Create an IPSec connection to the CPE object and provide required routing information.
3. **Have your network engineer configure your CPE device:** Your network engineer must configure your CPE device with information that Oracle provides during the previous steps. There is general information about the VCN, and specific information for each IPSec tunnel. This is the only part of the setup that you can't execute by using the Console or API. Without this configuration, traffic will not flow between your VCN and on-premises network. For more information, see Configuring Your CPE.
4. **Validate connectivity.**

Example: Setting Up a Proof of Concept IPSec VPN

This example scenario shows how to set up a single IPSec VPN with a simple layout that you might use for a proof of concept (POC). It follows tasks 1 and 2 in Overall Process and shows each component in the layout being created. For each task, there's a corresponding screenshot from the Console to help you understand what information is needed. For more complex layouts, see Example Layout with Multiple Geographic Areas or Example Layout with PAT.

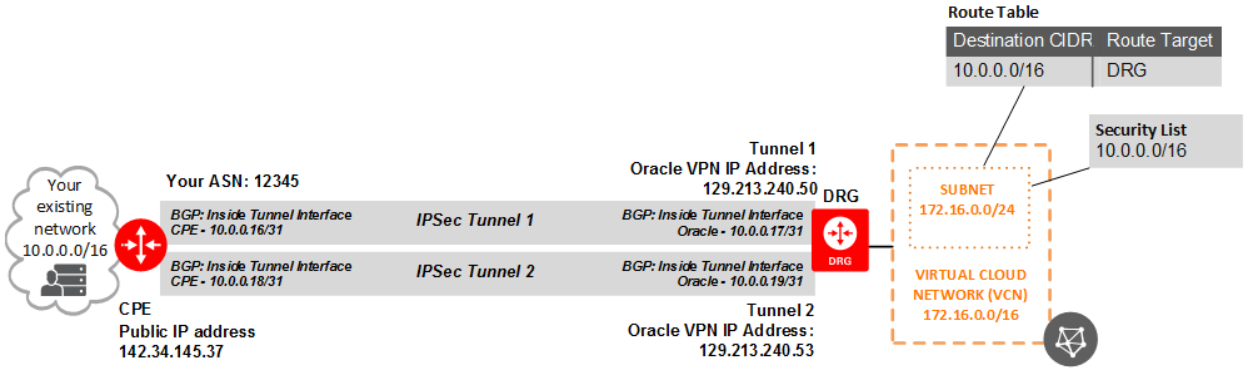
Task 1: Gather information

Question	Answer
What is your VCN's CIDR?	172.16.0.0/16
What is the public IP address of your CPE device? If you have multiple devices for redundancy, get the IP address for each. **Note:** If your CPE device is behind a NAT device, see If Your CPE Is Behind a NAT Device and also Information About Your CPE Device.	142.34.145.37
Will you be doing port address translation (PAT) between each CPE device and your VCN?	No

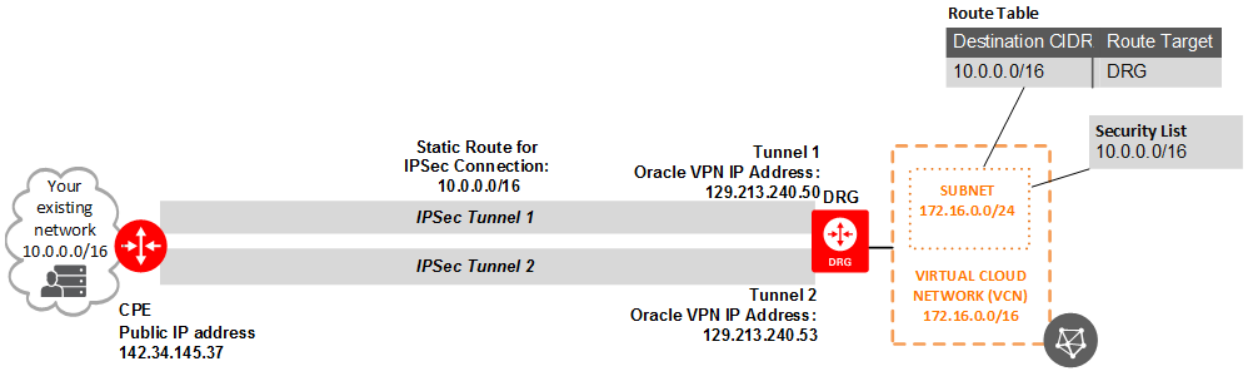
Question	Answer
What type of routing do you plan to use? If it's BGP dynamic routing, what are the BGP session IP addresses to use and the ASN of your network? If static routing, what are the static routes for your on-premises network? See Routing for the Oracle IPSec VPN.	**BGP dynamic routing example:** Tunnel 1: • BGP Inside tunnel interface - CPE: 10.0.0.16/31 • BGP Inside tunnel interface - Oracle: 10.0.0.17/31 Tunnel 2: • BGP Inside tunnel interface - CPE: 10.0.0.18/31 • BGP Inside tunnel interface - Oracle: 10.0.0.19/31 Network ASN:

Question	Answer
	12345 **Static routing example:** Use 10.0.0.0/16 for the static route for a simple POC.
Do you want to provide each tunnel's shared secret or let Oracle assign them? See About the Tunnel Shared Secret.	Let Oracle assign

Here's an example diagram for task 1 if you plan to use BGP dynamic routing:



Here's an example diagram for task 1 if you plan to use static routing:



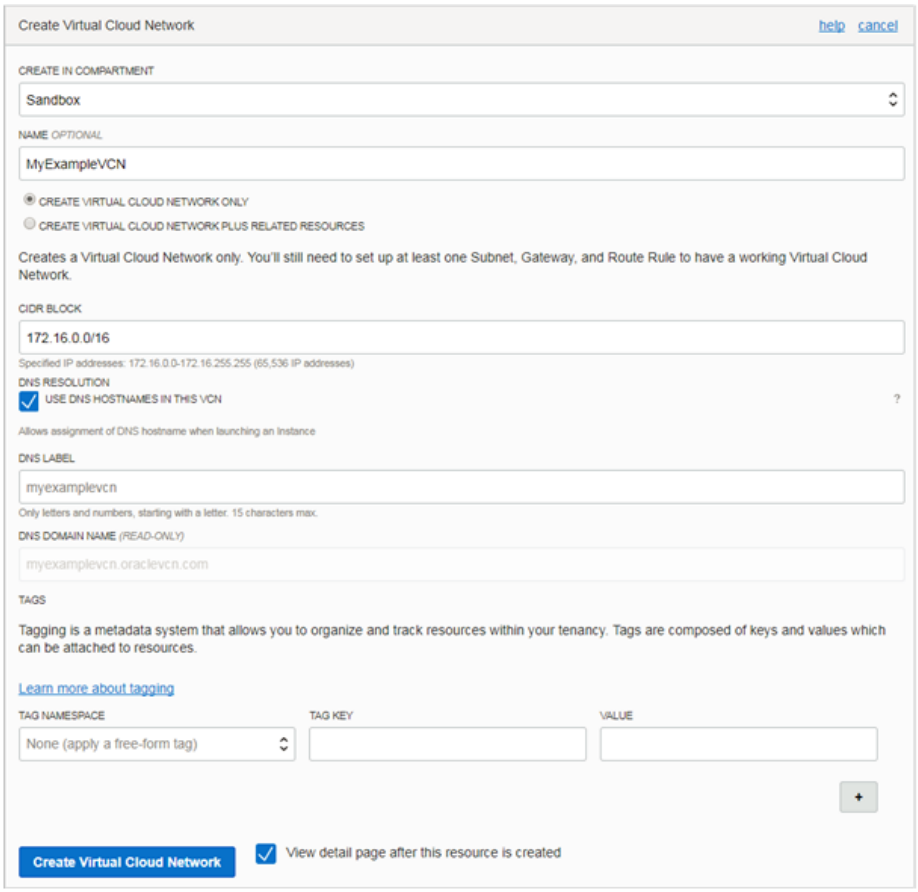
Task 2a: Create the VCN

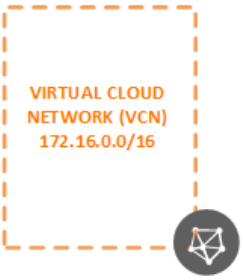
If you already have a VCN, skip to the next task.



Tip

When you use the Console to create a VCN, you can create only the VCN, or you can create the VCN with several related resources. This task creates only the VCN, and the subsequent tasks create the other required resources.





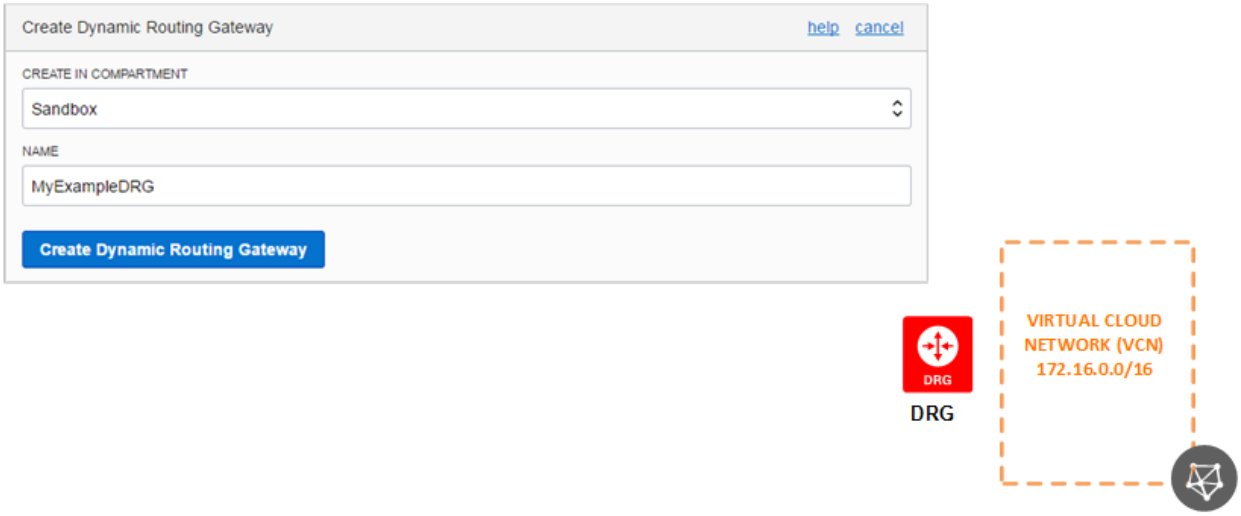
1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Choose a compartment you have permission to work in (on the left side of the page). The page updates to display only the resources in that compartment. If you're not sure which compartment to use, contact an administrator. For more information, see Access Control.
3. Click **Create Virtual Cloud Network**.
4. Enter the following values:

- **Create in Compartment:** Leave as is.
- **Name:** A descriptive name for the cloud network. It doesn't have to be unique, and it can't be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
- **Create Virtual Cloud Network Only:** Select this option.
- **CIDR Block:** A single, contiguous CIDR block for the cloud network (for example, 172.16.0.0/16). You *can't* change this value later. See Allowed VCN Size and Address Ranges. For reference, use a CIDR calculator.

5. You can provide values for the rest of the options, or you can ignore them:
   - **DNS Resolution:** If you want the instances in the VCN to have DNS hostnames (which can be used with the built-in DNS capability in the VCN), select the **Use DNS Hostnames in this VCN** check box. Then you can specify a DNS label for the VCN, or the Console will generate one for you. The dialog box automatically displays the corresponding **DNS Domain Name** for the VCN (*<VCN DNS label>*.oraclevcn.com). For more information, see DNS in Your Virtual Cloud Network.
   - **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
6. Click **Create Virtual Cloud Network**.

The VCN is created and displayed on the page. Ensure that it's done being provisioned before continuing.

Task 2b: Create the DRG

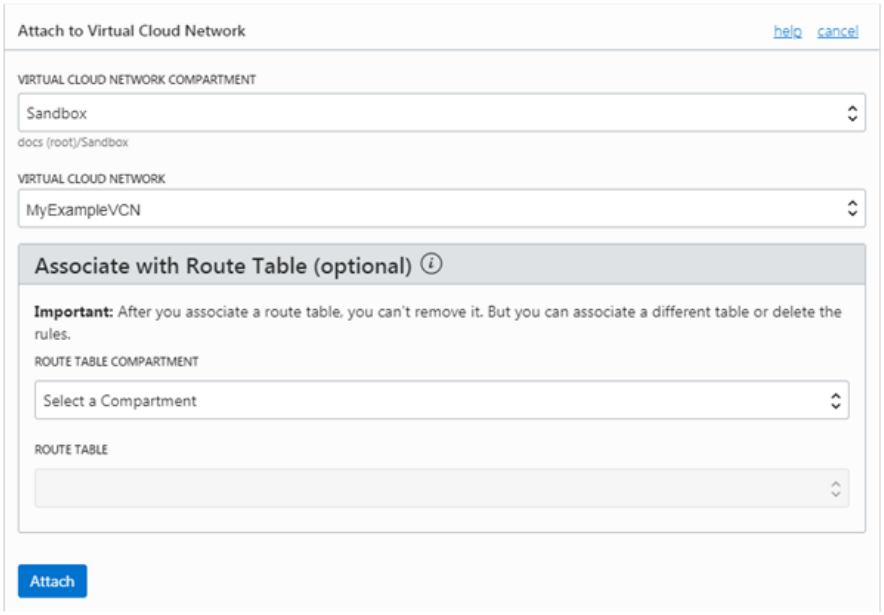


1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Dynamic Routing Gateways**.
2. Click **Create Dynamic Routing Gateway**.
3. Enter the following values:
   - **Create in Compartment:** Leave as is (the VCN's compartment).
   - **Name:** A descriptive name for the DRG. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
   - **Tags:** Leave as is. You can add tags later if you want. For more information, see Resource Tags.
4. Click **Create Dynamic Routing Gateway**.

The DRG is created and displayed on the page. Ensure that it's done being provisioned before continuing.

> **Tip**
>
> You could also use this DRG as the gateway for Oracle Cloud Infrastructure FastConnect, which is an alternative way to connect your on-premises network to your VCN.

Task 2c: Attach the DRG to the VCN



1. Click the name of the DRG that you just created.
2. On the left side of the page, click **Virtual Cloud Networks**.
3. Click **Attach to Virtual Cloud Network**.
4. Enter the following:

- **Virtual Cloud Network Compartment:** The compartment where the VCN resides.
- **Virtual Cloud Network:** The VCN you created earlier.
- **Associate with Route Table:** Do not select a route table here. You would associate a route table with a DRG attachment only if you're setting up an advanced scenario called transit routing.

5. Click **Attach**.

The attachment will be in the Attaching state for a short period before it's ready.

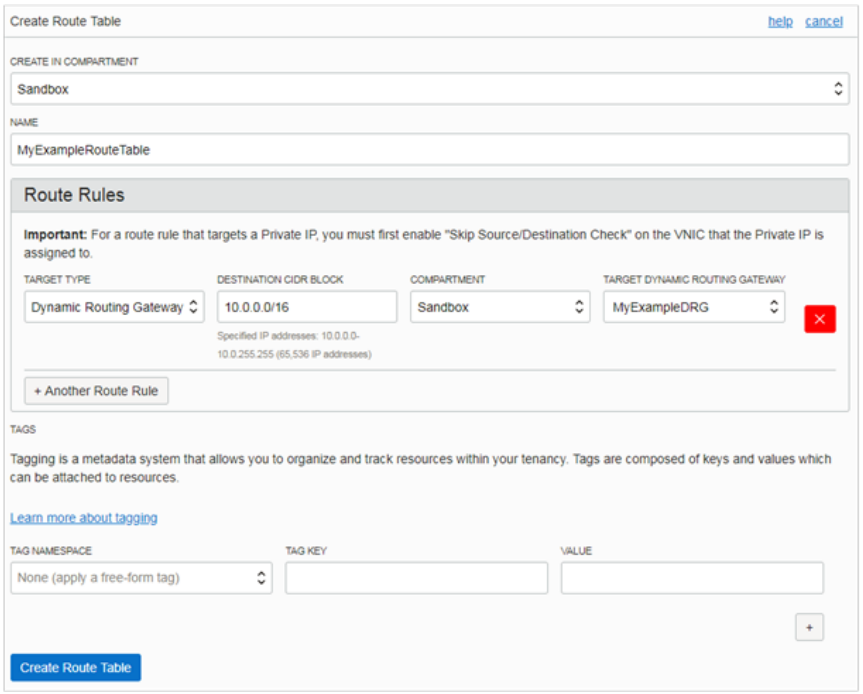
Task 2d: Create a route table and route rule for the DRG

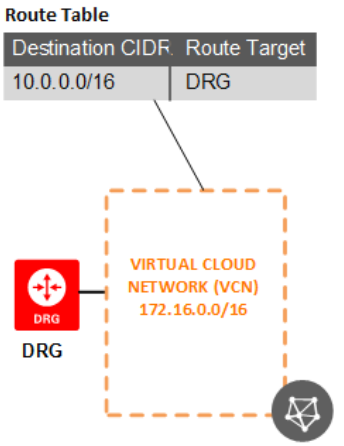
Although the VCN comes with a default route table (without any rules), in this task you create a custom route table with a route rule for the DRG. In this example, your on-premises network is 10.0.0.0/16. You create a route rule that takes any traffic destined for 10.0.0.0/16 and routes it to the DRG. When you create a subnet in task 2f, you associate this custom route table with the subnet.



Tip

If you already have an existing VCN with a subnet, you don't need to create a route table or subnet. Instead you can update the existing subnet's route table to include the route rule for the DRG.





1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click your VCN.
3. Click **Route Tables** to see your VCN's route tables.
4. Click **Create Route Table**.
5. Enter the following values:
    - **Name:** A descriptive name for the route table (for example, MyExampleRouteTable). The name doesn't have to be unique, and it can't be changed later in the Console (but you can change it in the API). Avoid entering confidential information.
    - **Create in compartment:** Leave as is.
    - Click **+ Additional Route Rule**, and enter the following:

- **Target Type:** Dynamic Routing Gateway. The VCN's attached DRG is automatically selected as the target, and you don't have to specify the target yourself.
- **Destination CIDR Block:** The CIDR for your on-premises network (10.0.0.0/16 in this example).

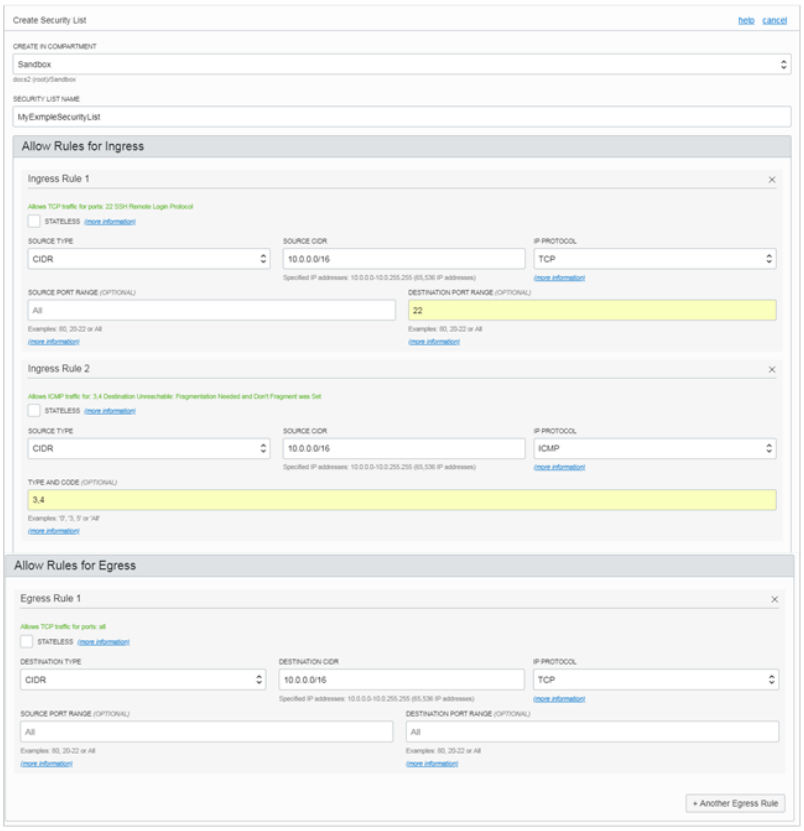
- **Tags:** Leave the tag information as is.

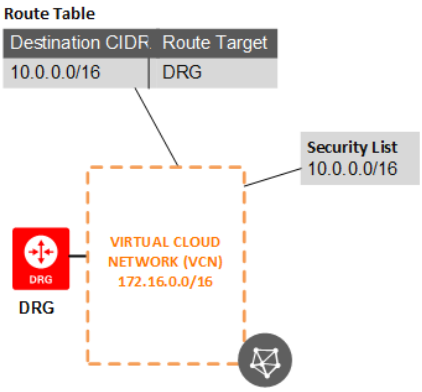
6. Click **Create Route Table**.

The route table is created and displayed on the page. However, the route table doesn't do anything unless you associate it with a subnet during subnet creation (see task 2f).

Task 2e: Create a security list

By default, incoming traffic to the instances in your VCN is set to DENY on all ports and all protocols. In this task, you set up two ingress rules and one egress rule to allow basic required network traffic. Your VCN comes with a default security list with a set of default rules. However, in this task you create a separate security list with a more restrictive set of rules focused on traffic with your on-premises network. When you create a subnet in task 2f, you associate this security list with the subnet.





Important

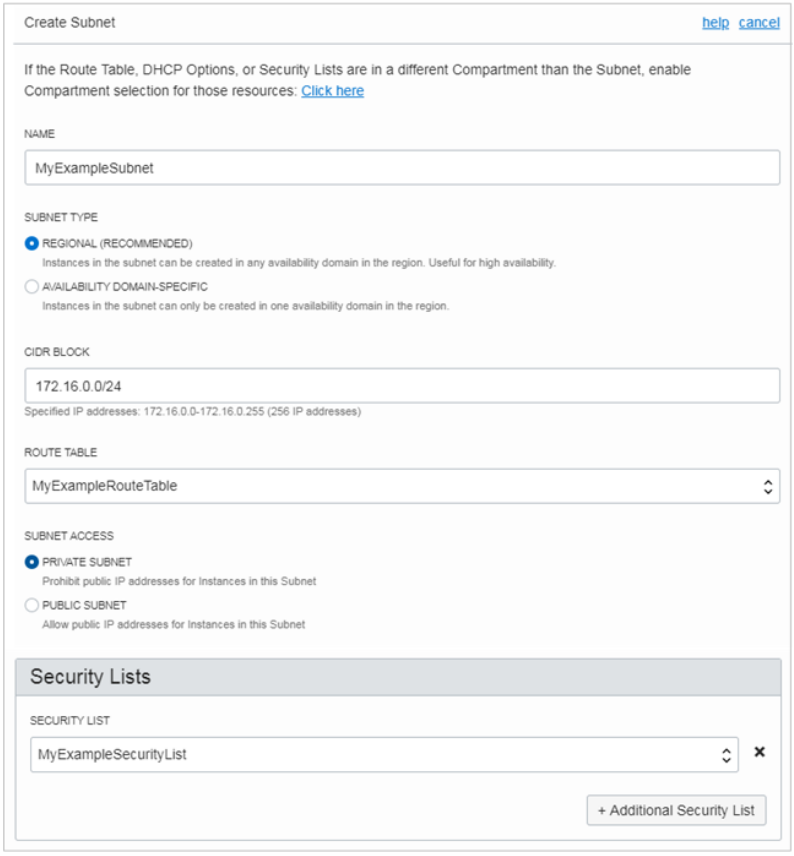
In the following procedure, ensure that the on-premises CIDR that you specify in the security list rules is the same (or smaller) than the CIDR that you specified in the route rule in the preceding task. Otherwise, traffic will be blocked by the security lists.

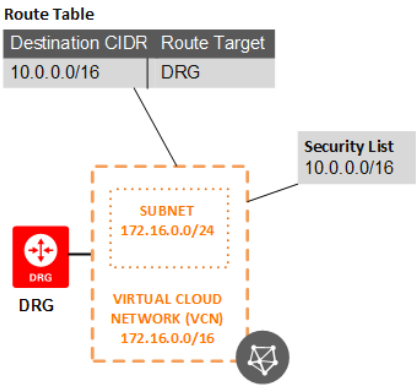
1. While still viewing your VCN, click **Security Lists** on the left side of the page.
2. Click **Create Security List**.
3. Enter the following values:
    - **Name:** A descriptive name for the security list. The name doesn't have to be unique, and it cannot be changed later in the Console (but you can change it in the API). Avoid entering confidential information.
    - **Create in compartment:** Leave as is.
4. In the **Allow Rules for Ingress** section, click **Add Ingress Rule** and enter the following values for the ingress rule, which allows incoming SSH on TCP port 22 from your on-premises network:
    - **Source Type:** CIDR
    - **Source CIDR:** The CIDR for your on-premises network (10.0.0.0/16 in this example)
    - **IP Protocol:** TCP.
    - **Source Port Range:** Leave as is (the default All).
    - **Destination Port Range:** 22 (for SSH traffic).
5. In the **Allow Rules for Egress** section, click **Add Egress Rule** enter the following values for the egress rule, which allows outgoing TCP traffic on all ports to your on-premises network:
    - **Destination Type:** CIDR
    - **Destination CIDR:** The CIDR for your on-premises network (10.0.0.0/16 in this example).
    - **IP Protocol:** TCP.
    - **Source Port Range:** Leave as is (the default All).
    - **Destination Port Range:** Leave as is (the default All).
6. Leave the tag information as is.
7. Click **Create Security List**.

The security list is created and displayed on the page. However, the security list doesn't do anything unless you associate it with a subnet during subnet creation (see task 2f).

Task 2f: Create a subnet

In this task, you create a subnet in the VCN. Typically a subnet has a CIDR block smaller than the VCN's CIDR. Any instances that you create in this subnet have access to your on-premises network. Oracle recommends using regional subnets. Here you create a regional private subnet.



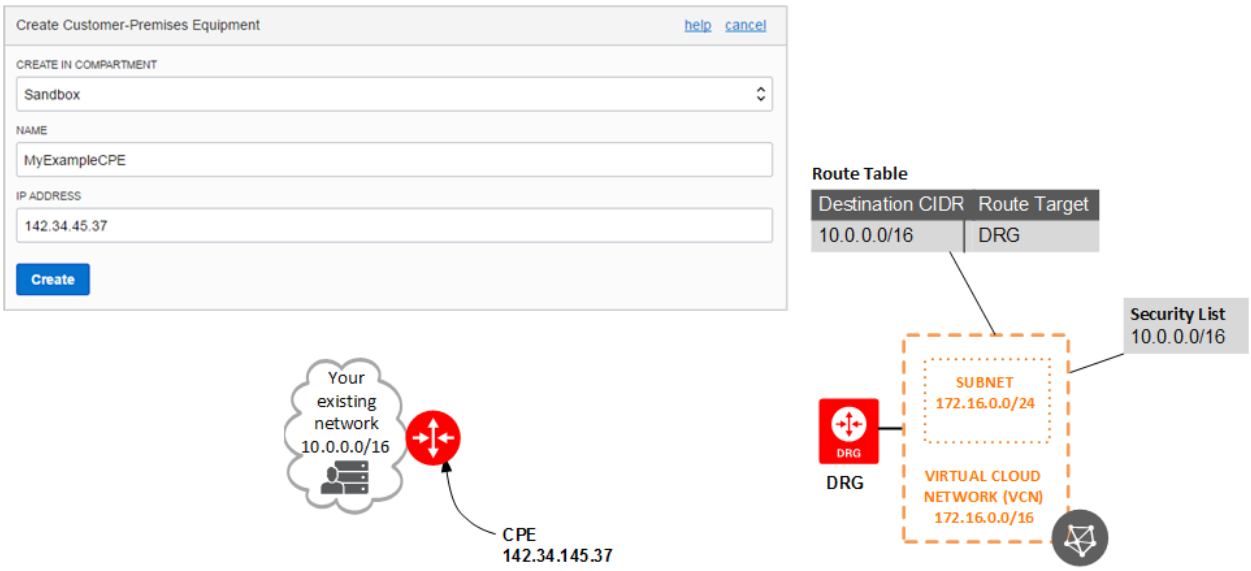


1. While still viewing your VCN, click **Subnets** on the left side of the page.
2. Click **Create Subnet**.
3. Enter the following values:
    - **Name:** A descriptive name for the subnet. It doesn't have to be unique, and it can't be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
    - **Regional or AD-specific subnet:** Select the radio button for **Regional**. Oracle recommends using regional subnets.
    - **CIDR Block:** A single, contiguous CIDR block for the subnet (for example, 172.16.0.0/24). It must be within the cloud network's CIDR block and can't overlap with any other subnets. You *can't* change this value later. See Allowed VCN Size and Address Ranges. For reference, use a CIDR calculator.
    - **Route Table:** The route table that you created earlier.
    - **Private Subnet**: Select this option. For more information, see Access to the Internet.
    - **Use DNS Hostnames in this Subnet:** Leave as is (selected).
    - **DHCP Options:** The set of DHCP options to associate with the subnet. Select the default set of DHCP options for the VCN.
    - **Security Lists:** The security list that you created earlier.
    - **Tags:** Leave as is. You can add tags later if you want. For more information, see Resource Tags.
4. Click **Create Subnet**.

The subnet is created and displayed on the page. The basic VCN in this example is now set up, and you're ready to create the remaining components for the IPSec VPN.

Task 2g: Create a CPE object and provide your CPE device's public IP address

In this task, you create the CPE object, which is a virtual representation of your CPE device.



1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Customer-Premises Equipment**.
2. Click **Create Customer-Premises Equipment**.
3. Enter the following values:
   - **Create in Compartment:** Leave as is (the VCN's compartment).
   - **Name:** A descriptive name for the CPE object. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
   - **IP Address:** The public IP address of the CPE device at your end of the VPN (see the list of information to gather in Before You Get Started).
   - **Tags:** Leave as is. You can add tags later if you want. For more information, see Resource Tags.
4. Click **Create**.

The CPE object is created and displayed on the page.

Task 2h: Create an IPSec connection to the CPE object

In this task, you create the IPSec tunnels and configure the type of routing for them (BGP dynamic routing or static routing).

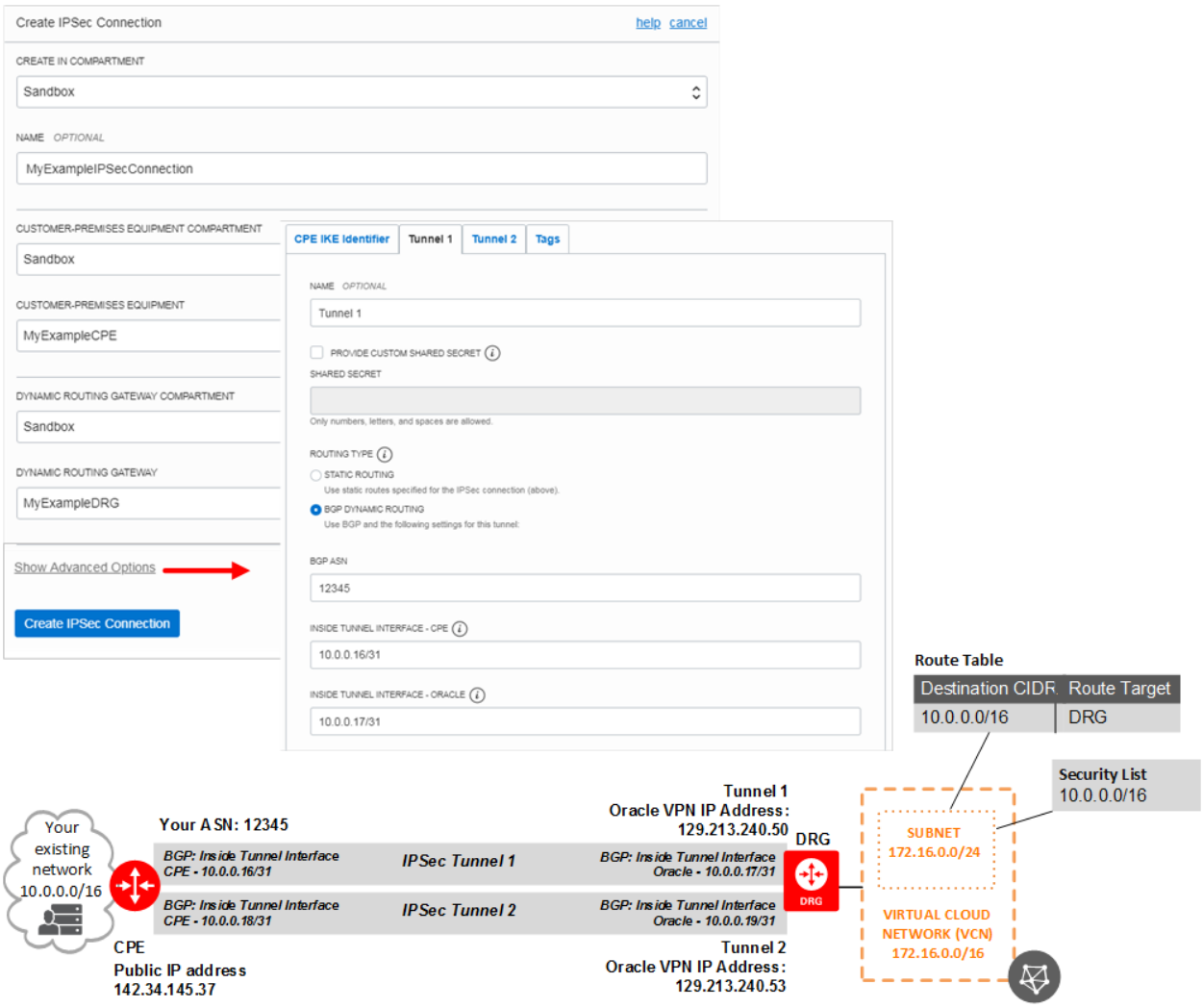


Tip

If you have an existing Oracle IPSec VPN that uses static routing, you can change the tunnels to instead use BGP dynamic routing.

For BGP dynamic routing

In this example, you configure both tunnels to use BGP dynamic routing.



1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **IPSec Connections**.
2. Click **Create IPSec Connection**.
3. Enter the following values:

- **Create in Compartment:** Leave as is (the VCN's compartment).
- **Name:** Enter a descriptive name for the IPSec connection. It doesn't have to be unique, and you can change it later if you like. Avoid entering confidential information.
- **Customer-Premises Equipment Compartment:** Leave as is (the VCN's compartment).
- **Customer-Premises Equipment:** Select the CPE object that you created earlier.
- **Dynamic Routing Gateway Compartment:** Leave as is (the VCN's compartment).
- **Dynamic Routing Gateway:** Select the DRG that you created earlier.
- **Static Route CIDR:** Leave empty because this IPSec connection uses BGP dynamic routing. You configure the two tunnels to use BGP in subsequent steps.

4. Click **Show Advanced Options**.
5. On the **CPE IKE Identifier** tab (optional): Oracle defaults to using the public IP address of the CPE. But if your CPE is behind a NAT device, you might need to enter a different value. You can either enter the new value here, or change the value later.
6. On the **Tunnel 1** tab (required):
   - **Name:** Enter a descriptive name for the tunnel. It doesn't have to be unique, and you can change it later if you like. Avoid entering confidential information.
   - **Provide custom shared secret** (optional): By default, Oracle provides the shared secret for the tunnel. If you want to provide it instead, select this check box and enter the shared secret. You can change the shared secret later.
   - **Routing Type:** Select the radio button for **BGP Dynamic Routing**.
   - **BGP ASN:** Enter your network's ASN.

- **Inside Tunnel Interface - CPE:** Enter the BGP IP address with subnet mask (either /30 or /31) for the CPE end of the tunnel. For example: 10.0.0.16/31. The IP address must be part of the IPSec VPN's encryption domain.
- **Inside Tunnel Interface - Oracle:** Enter the BGP IP address with subnet mask (either /30 or /31) for the Oracle end of the tunnel. For example: 10.0.0.17/31. The IP address must be part of the IPSec VPN's encryption domain.

7. On the **Tunnel 2** tab (required):
   - **Name:** Enter a descriptive name for the tunnel. It doesn't have to be unique, and you can change it later if you like. Avoid entering confidential information.
   - **Provide custom shared secret** (optional): By default, Oracle provides the shared secret for the tunnel. If you want to provide it instead, select this check box and enter the shared secret. You can change the shared secret later.
   - **Routing Type:** Select the radio button for **BGP Dynamic Routing**.
   - **BGP ASN:** Enter your network's ASN.
   - **Inside Tunnel Interface - CPE:** Enter the BGP IP address with subnet mask (either /30 or /31) for the CPE end of the tunnel. Use a different IP address than for tunnel 1. For example: 10.0.0.18/31. The IP address must be part of the IPSec VPN's encryption domain.
   - **Inside Tunnel Interface - Oracle:** Enter the BGP IP address with subnet mask (either /30 or /31) for the Oracle end of the tunnel. Use a different IP address than for tunnel 1. For example: 10.0.0.19/31. The IP address must be part of the IPSec VPN's encryption domain.
8. On the **Tags** tab (optional): Leave as is. You can add tags later if you want. For more information, see Resource Tags.
9. Click **Create IPSec Connection**.
   The IPSec connection is created and displayed on the page. It will be in the Provisioning state for a short period.
   The displayed tunnel information includes:

- The Oracle VPN IP address (for the Oracle VPN headend).
- The tunnel's IPSec status (possible values are Up, Down, and Down for Maintenance). At this point, the status is Down. Your network engineer still must configure your CPE device.
- The tunnel's BGP status. At this point, the status is Down. Your network engineer still must configure your CPE device.

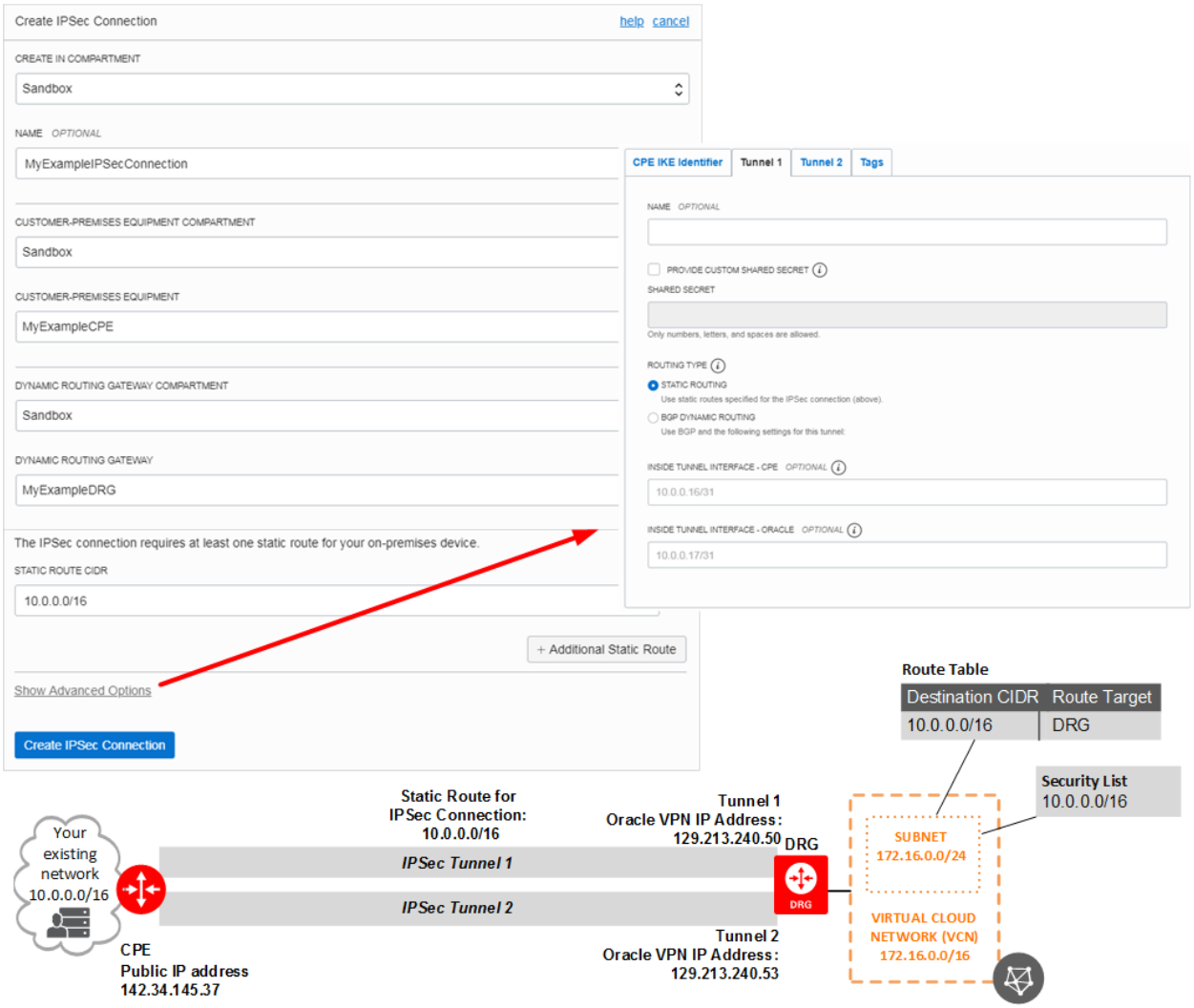
To view the tunnel's shared secret, click the tunnel to view its details, and then click **Show** next to **Shared Secret**.

10. Copy the Oracle VPN IP address and shared secret for each of the tunnels to an email or other location so you can deliver it to the network engineer who will configure your CPE device.

    You can view this tunnel information here in the Console at any time.

You have now created all the components required for the IPSec VPN. But your network engineer must configure your CPE device before network traffic can flow between your on-premises network and VCN.

For static routing



1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **IPSec Connections**.

2. Click **Create IPSec Connection**.
3. Enter the following values:
    - **Create in Compartment:** Leave as is (the VCN's compartment).
    - **Name:** Enter a descriptive name for the IPSec connection. It doesn't have to be unique, and you can change it later if you like. Avoid entering confidential information.
    - **Customer-Premises Equipment Compartment:** Leave as is (the VCN's compartment).
    - **Customer-Premises Equipment:** Select the CPE object that you created earlier.
    - **Dynamic Routing Gateway Compartment:** Leave as is (the VCN's compartment).
    - **Dynamic Routing Gateway:** Select the DRG that you created earlier.
    - **Static Route CIDR:** Enter at least one static route CIDR (see the list of information to gather in Before You Get Started). For this example, enter 10.0.0.0/16. You can enter up to 10 static routes, and you can change the static routes later if you like.
4. Click **Show Advanced Options**.
5. On the **CPE IKE Identifier** tab (optional): Oracle defaults to using the public IP address of the CPE. But if your CPE is behind a NAT device, you might need to enter a different value. You can either enter the new value here, or change the value later.
6. On the **Tunnel 1** tab (optional):
    - **Tunnel Name:** Enter a descriptive name for the tunnel. It doesn't have to be unique, and you can change it later if you like. Avoid entering confidential information.
    - **Provide custom shared secret** (optional): By default, Oracle provides the shared secret for the tunnel. If you want to provide it instead, select this check box and enter the shared secret. You can change the shared secret later.

- **Routing Type:** Leave the radio button for **Static Routing** selected.
- **Inside Tunnel Interface - CPE** (optional): You can provide an IP address with subnet mask (either /30 or /31) for the CPE end of the tunnel for the purposes of tunnel troubleshooting or monitoring. For example: 10.0.0.16/31. The IP address must be part of the IPSec VPN's encryption domain.
- **Inside Tunnel Interface - Oracle** (optional): You can provide an IP address with subnet mask (either /30 or /31) for the Oracle end of the tunnel for the purposes of tunnel troubleshooting or monitoring. For example: 10.0.0.17/31. The IP address must be part of the IPSec VPN's encryption domain.

7. On the **Tunnel 2** tab (optional):
   - **Tunnel Name:** Enter a descriptive name for the tunnel. It doesn't have to be unique, and you can change it later if you like. Avoid entering confidential information.
   - **Provide custom shared secret** (optional): By default, Oracle provides the shared secret for the tunnel. If you want to provide it instead, select this check box and enter the shared secret. You can change the shared secret later.
   - **Routing Type:** Leave the radio button for **Static Routing** selected.
   - **Inside Tunnel Interface - CPE** (optional): You can provide an IP address with subnet mask (either /30 or /31) for the CPE end of the tunnel for the purposes of tunnel troubleshooting or monitoring. Use a different IP address than for tunnel 1. For example: 10.0.0.18/31. The IP address must be part of the IPSec VPN's encryption domain.
   - **Inside Tunnel Interface - Oracle** (optional): You can provide an IP address with subnet mask (either /30 or /31) for the Oracle end of the tunnel for the purposes of tunnel troubleshooting or monitoring. Use a different IP address than for tunnel 1. For example: 10.0.0.19/31. The IP address must be part of the IPSec VPN's encryption domain.
8. **Tags:** Leave as is. You can add tags later if you want. For more information, see Resource Tags.

9. Click **Create IPSec Connection**.
   The IPSec connection is created and displayed on the page. It will be in the Provisioning state for a short period.
   The displayed tunnel information includes:
   - The Oracle VPN IP address (for the Oracle VPN headend).
   - The tunnel's IPSec status (possible values are Up, Down, and Down for Maintenance). At this point, the status is Down. Your network engineer still must configure your CPE device.

   To view the tunnel's shared secret, click the tunnel to view its details, and then click **Show** next to **Shared Secret**.
10. Copy the Oracle VPN IP address and shared secret for each of the tunnels to an email or other location so you can deliver it to the network engineer who will configure the CPE device.
    You can view this tunnel information here in the Console at any time.

You have now created all the components required for the IPSec VPN. But your network engineer must configure the CPE device before network traffic can flow between your on-premises network and VCN.

For more information, see Configuring Your CPE.

Task 3: Have your network engineer configure your CPE

Provide your network engineer with the following information:

- The VCN's OCID.
- The VCN's CIDR and subnet mask.
- The shared secret and Oracle VPN IP address for each tunnel (from task 2h).
- If using BGP: the BGP session information for each tunnel, which includes the IP addresses and Oracle ASN (31898).

- The information required to configure your particular CPE device, which is listed in the topic for each verified type of CPE device.
- The general IPSec parameters that Oracle supports.
- A link to this topic: Configuring Your CPE.



Important

Be sure to have your network engineer configure your CPE device to support both of the tunnels in case one fails or Oracle takes one down for maintenance. If you're using BGP, see Important Routing Details for an Oracle IPSec VPN.

Task 4: Validate connectivity

After the network engineer configures your CPE device, you can confirm that the tunnel's IPSec status is Up and green. Next, you can create a Linux instance in the subnet in your VCN. You should then be able to use SSH to connect to the instance's private IP address from a host in your on-premises network. For more information, see Creating an Instance.

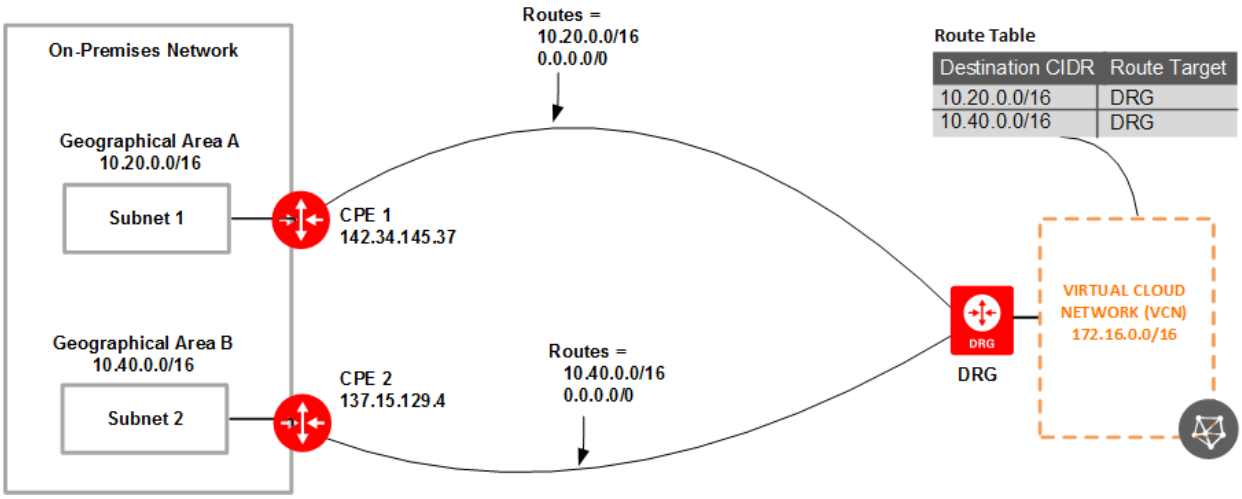
Example Layout with Multiple Geographic Areas

The following diagram shows an example with this configuration:

- Two networks in separate geographical areas that each connect to your VCN
- A single CPE device in each area
- Two IPSec VPNs (one for each CPE device)

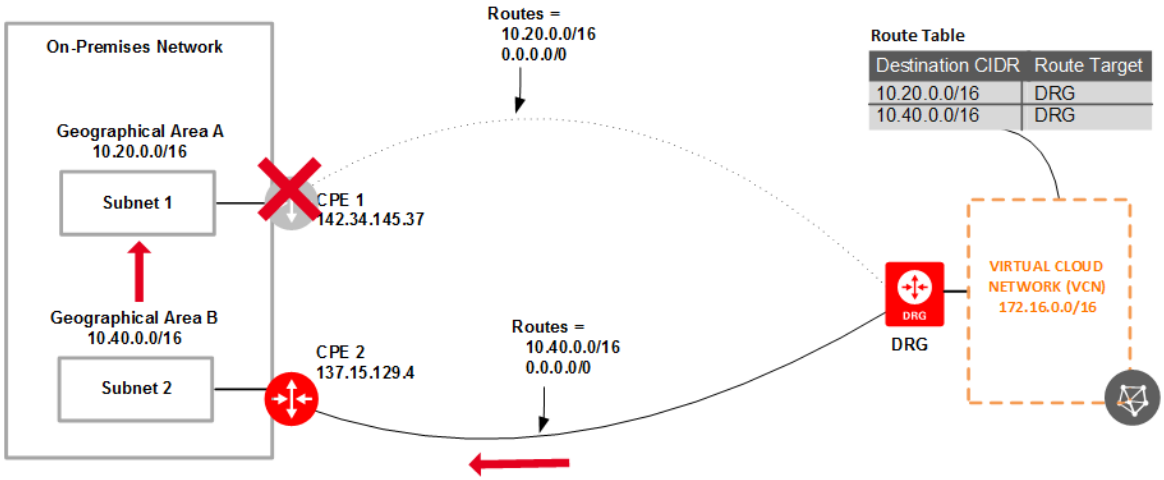
Notice that each IPSec VPN has two routes associated with it: one for the particular geographical area's subnet, and a default 0.0.0.0/0 route. Oracle learns about the available

routes for each tunnel either through BGP (if the tunnels use BGP), or because you've set them as static routes for the IPSec connection (if the tunnels use static routing).

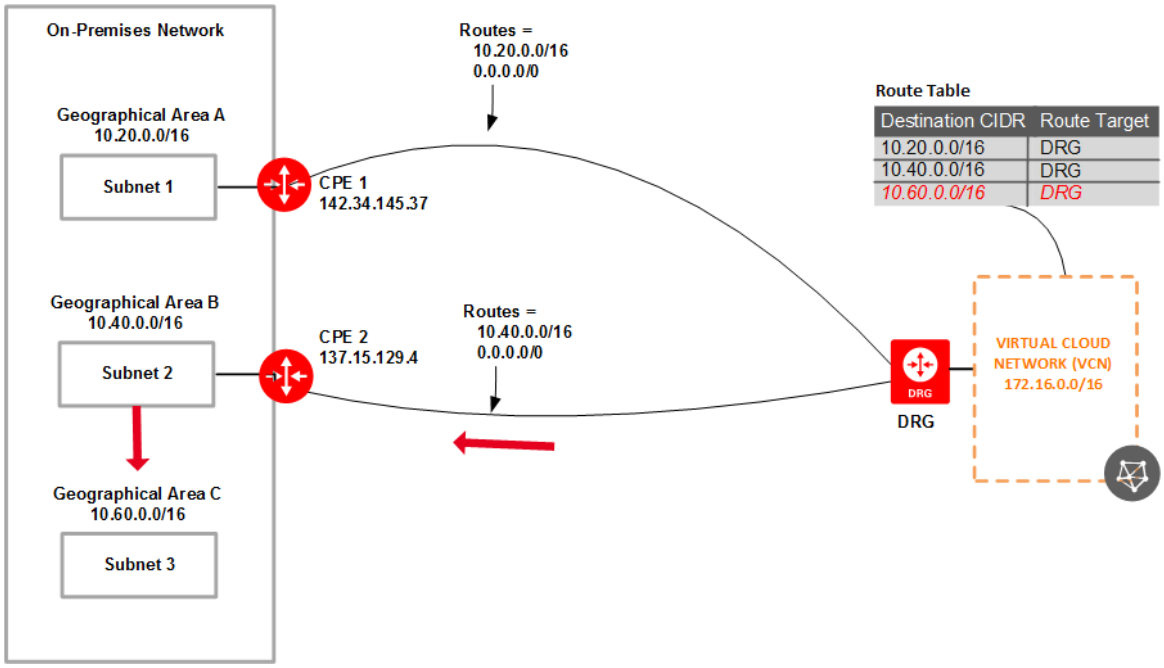


Following are some examples of situations in which the 0.0.0.0/0 route can provide flexibility:

- Assume that the CPE 1 device goes down (see the next diagram). If Subnet 1 and Subnet 2 can communicate with each other, your VCN could still reach the systems in Subnet 1 because of the 0.0.0.0/0 route that goes to CPE 2.



- Assume that your organization adds a new geographical area with Subnet 3 and initially just connects it to Subnet 2 (see the next diagram). If you added a route rule to your VCN's route table for Subnet 3, the VCN could reach systems in Subnet 3 because of the 0.0.0.0/0 route that goes to CPE 2.



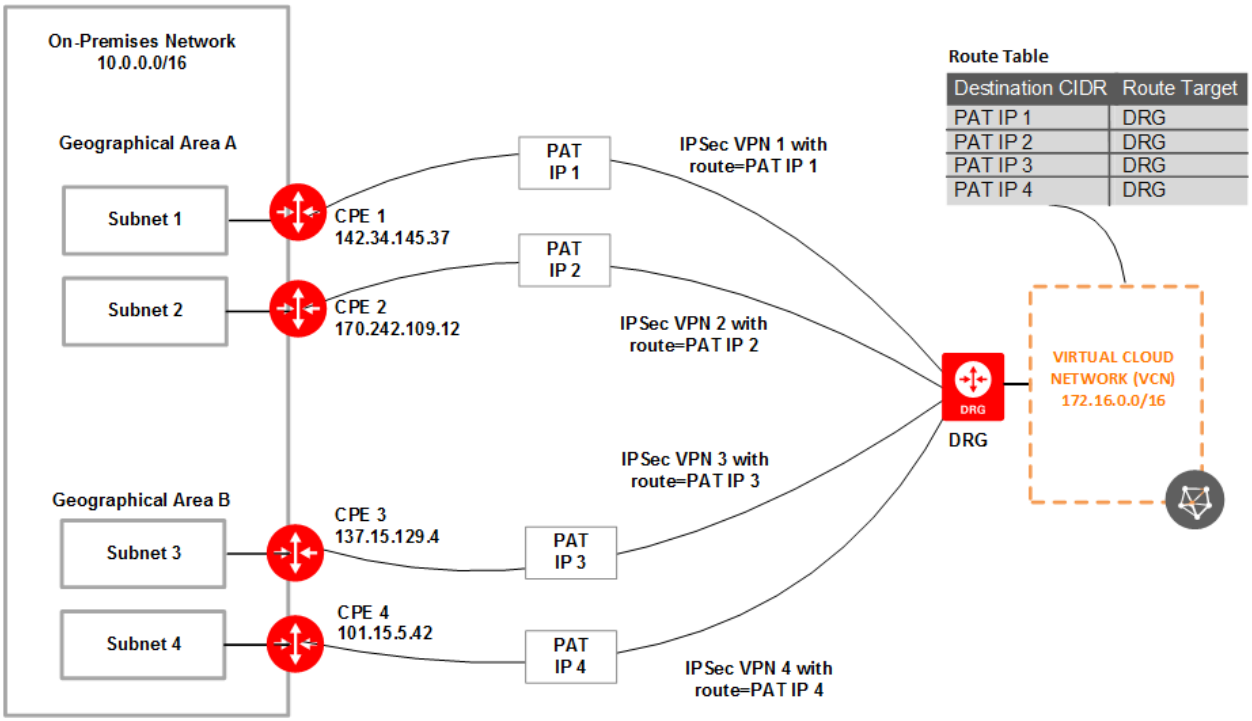
Example Layout with PAT

The following diagram shows an example with this configuration:

- Two networks in separate geographical areas that each connect to your VCN
- Redundant CPE devices (two in each geographical area)
- Four IPSec VPNs (one for each CPE device)
- Port address translation (PAT) for each CPE device

For each of the four IPSec connections, the route that Oracle needs to know about is the PAT IP address for the specific CPE device. Oracle learns about the PAT IP address route for each tunnel either through BGP (if the tunnels use BGP), or because you've set the relevant address as a static route for the IPSec connection (if the tunnels use static routing).

When you set up the route rules for the VCN, you specify a rule for each PAT IP address (or an aggregate CIDR that covers them all) with your DRG as the rule's target.



What's Next?

See these related topics and procedures:

- Configuring Your CPE
- Verified CPE Devices

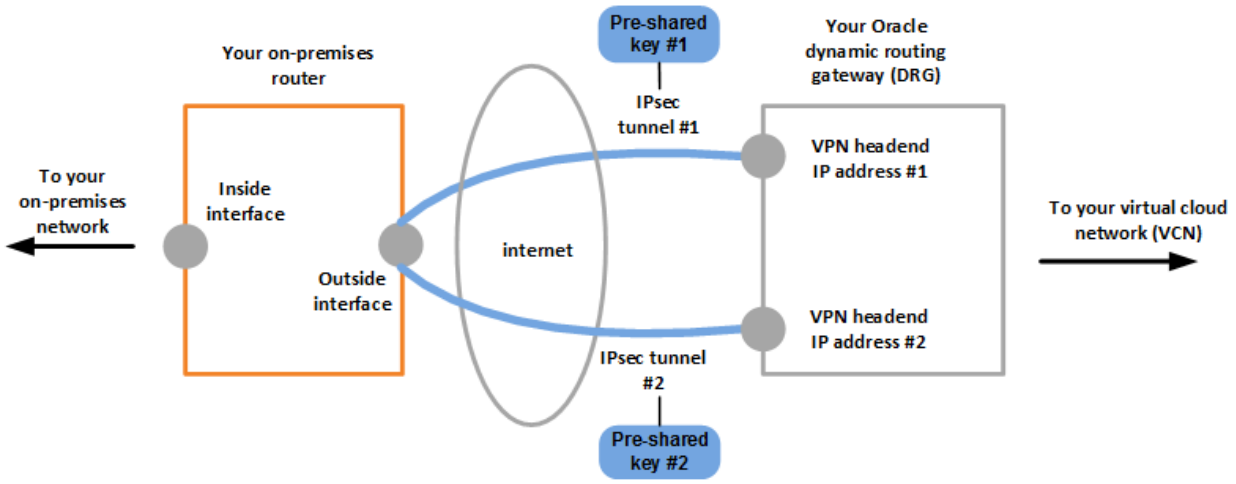
- Changing from Static Routing to BGP Dynamic Routing
- Working with VPN Connect
- VPN Connect FAQ
- Using the API for VPN Connect

Configuring Your CPE

This topic is for network engineers. It explains how to configure the on-premises device (the customer-premises equipment, or CPE) at your end of the IPSec VPN so traffic can flow between your on-premises network and virtual cloud network (VCN). See these related topics:

- Overview of Networking: For general information about the parts of a VCN
- VPN Connect: For various topics about IPSec VPNs
- Verified CPE Devices: For a list of CPE devices Oracle has verified

The following figure shows the basic layout of the IPSec VPN connection.



Requirements and Prerequisites

There are several requirements and prerequisites to be aware of before moving forward.

Routing Considerations

For important details about routing for your IPSec VPN see Routing for the Oracle IPSec VPN.

Oracle uses asymmetric routing across the multiple tunnels that make up the IPSec VPN connection. Even if you configure one tunnel as primary and another as backup, traffic from your VCN to your on-premises network can use any tunnel that is "up" on your device. Configure your firewalls accordingly. Otherwise, ping tests or application traffic across the connection will not reliably work.

If you use BGP dynamic routing with your IPSec VPN, you can configure routing so that Oracle prefers one tunnel over the other.

Note that the Cisco ASA policy-based configuration uses a single tunnel.

Creation of Cloud Network Components

You or someone in your organization must have already used the Oracle Console to create a VCN and an IPSec connection, which consists of multiple IPSec tunnels for redundancy. You must gather the following information about those components:

- VCN OCID: The VCN OCID is a unique Oracle Cloud Infrastructure identifier that has a UUID at the end. You can use this UUID or any other string that helps you identify this VCN in the device configuration and doesn't conflict with other object-group or access-list names.
- VCN CIDR
- VCN CIDR subnet mask
- For each IPSec tunnel:
    - The IP address of the Oracle IPSec tunnel endpoint (the VPN headend)
    - The shared secret

Information About Your CPE Device

You also need some basic information about the inside and outside interfaces of your on-premises device (your CPE). For a list of the required information for your particular CPE, see

the links in this list: Verified CPE Devices.

Oracle recommends that you disable NAT-T at your CPE when establishing IPSec tunnels with Oracle Cloud Infrastructure. Unless you have multiple CPEs sharing the same NAT IP, NAT-T is not required.

If your CPE is behind a NAT device, you can provide Oracle with your CPE's IKE identifier. For more information, see If Your CPE Is Behind a NAT Device.

Route-Based Versus Policy-Based IPSec

The IPSec protocol uses Security Associations (SAs) to determine how to encrypt packets. Within each SA, you define Security Parameter Indexes (SPIs) to map a packet's source and destination IP address and protocol type to an entry in the SA database to define how to encrypt or decrypt a packet. Note that other vendors or industry documentation might use the term *Encryption Domain* instead of *SPIs*.

There are two general methods of implementing IPSec tunnels:

- **route-based tunnels:** Also called *next-hop-based tunnels*. A route table lookup is performed on a packet's destination IP address. If that route's egress interface is an IPSec tunnel, the packet is encrypted and sent to the other end of the tunnel.
- **policy-based tunnels:** The packet's source and destination IP address and protocol are matched against a list of policy statements. If a match is found, the packet is encrypted based on the rules in that policy statement.

The Oracle VPN headends use route-based tunnels, but can work with policy-based tunnels with some caveats listed in the following section.

Oracle supports only a single SPI encryption domain.

SPI for route-based tunnels

If your CPE supports route-based tunnels, use that method to configure the tunnel. It's the simplest configuration with the most interoperability with the Oracle VPN headend.

Route-based IPSec uses an SPI with the following values:

- **Source IP address:** Any (0.0.0.0/0)
- **Destination IP address:** Any (0.0.0.0/0)
- **Protocol:** IPv4

SPI for policy-based tunnels

If your CPE supports only policy-based tunnels, there are restrictions on the policy that you can use on the CPE.

The Oracle VPN headend supports only a single SPI encryption domain. If your policy includes multiple entries, the tunnel will bounce or there will be connectivity problems in which only a single policy works at any one time.

If you use policy-based IPSec, Oracle recommends using a *single SPI* with the following values:

- **Source IP address:** Any (0.0.0.0/0)
- **Destination IP address:** VCN CIDR (example: 10.120.0.0/20)
- **Protocol: IPv4**

Make sure the single SPI matches any traffic that needs to go from your on-premises network across the IPSec tunnel to the VCN. The VCN CIDR must not overlap with your on-premises network.

IPSec VPN Best Practices

- **Configure all tunnels for every IPSec connection:** Oracle deploys multiple IPSec headends for all your connections to provide high availability for your mission-critical workloads. Configuring all the available tunnels is a key part of the "Design for Failure" philosophy. (Exception: Cisco ASA policy-based configuration, which uses a single tunnel.)

- **Have redundant CPEs in your on-premises locations:** Each of your sites that connects with IPSec to Oracle Cloud Infrastructure should have redundant CPE devices. You add each CPE to the Oracle Cloud Infrastructure Console and create a separate IPSec connection between your dynamic routing gateway (DRG) and each CPE. For each IPSec connection, Oracle provisions two tunnels on geographically redundant IPSec headends. Oracle may use any tunnel that is "up" to send traffic back to your on-premises network. For more information, see Routing for the Oracle IPSec VPN.
- **Consider backup aggregate routes:** If you have multiple sites connected via IPSec VPNs to Oracle Cloud Infrastructure, and those sites are connected to your on-premises backbone routers, consider configuring your IPSec connection routes with both the local site aggregate route as well as a default route.

  Note that the DRG routes learned from the IPSec connections are only used by traffic you route from your VCN to your DRG. The default route will only be used by traffic sent to your DRG whose destination IP address does not match the more specific routes of any of your tunnels.

Confirming the Status of the Connection

After you configure the IPSec connection, you can test the connection by launching an instance into the VCN and then pinging it from your on-premises network. For information about launching an instance, see Launching an Instance.

You can get the status of the IPSec tunnels in the API or Console. For instructions, see To view the status and configuration information for the IPSec tunnels.

Device Configurations

For links to the specific configuration information for each verified CPE device, see Verified CPE Devices.

Generic CPE Configuration Information

Oracle Cloud Infrastructure VPN service uses standards-based IPSec encryption. If your CPE device is not one that already has configuration information (see Device Configurations), use the information here to configure your device.



Important

Oracle uses asymmetric routing across the multiple tunnels that make up the IPSec VPN connection. Even if you configure one tunnel as primary and another as backup, traffic from your VCN to your on-premises network can use any tunnel that is "up" on your device. Configure your firewalls accordingly. Otherwise, ping tests or application traffic across the connection will not reliably work.

ISAKMP Policy Options

- ISAKMP Protocol version 1
- Exchange type: Main mode
- Authentication method: pre-shared-keys
- Encryption: *AES-256-cbc*, AES-192-cbc, AES-128-cbc
- Authentication algorithm: *SHA-384*, SHA-256, SHA1 (also called SHA or SHA1-96)
- Diffie-Hellman group: *group 5*, group 2, group 1
- IKE session key lifetime: 28800 seconds (8 hours)

IPSec Policy Options

- IPSec protocol: ESP, tunnel-mode
- Encryption: *AES-256-cbc*, AES-192-cbc, AES-128-cbc
- Authentication algorithm: HMAC-SHA1-96
- IPSec session key lifetime: 3600 seconds (1 hour)
- Perfect Forward Secrecy (PFS): enabled, group 5

Security Parameter Index

The values for the Security Parameter Index (SPI) depend on whether your CPE supports route-based tunnels or policy-based tunnels. For more information about the correct SPI values to use, see Route-Based Versus Policy-Based IPSec.

Check Point

This section includes two different sets of instructions: for domain-based tunnel configuration and VPN tunnel interface (VTI) configuration.

This configuration was validated using a Check Point 2200.



Important

Oracle uses asymmetric routing across the multiple tunnels that make up the IPSec VPN connection. Even if you configure one tunnel as primary and another as backup, traffic from your VCN to your on-premises network can use any tunnel that is "up" on your device. Configure your firewalls accordingly. Otherwise, ping tests or application traffic across the connection will not reliably work.

Parameters from API or Console

Get the following parameters from the Oracle Cloud Infrastructure Console or API.

${ipAddress#}

- Oracle VPN headend IPSec tunnel endpoints. There is one value for each tunnel.
- Example value: 129.146.12.52

${sharedSecret#}

- The IPSec IKE pre-shared-key. There is one value for each tunnel.
- Example value: EXAMPLEDPfAMkD7nTH3SWr6OFabdT6exXn6enSlsKbE

Additional Configuration Parameters

The Check Point config requires the following additional variables:

${cpePublicInterface}

- The name of the Interface where the CPE's public IP address is configured.
- Example Value: eth1

${VcnCidrBlock}

- When creating the VCN, your company selected this CIDR to represent the IP aggregate network for all VCN hosts.
- Example Value: 10.0.0.0/16

${VcnCidrNetwork}** and **${VcnCidrNetmask}

- These are the base address and netmask for the `${VcnCidrBlock}`
- For more information, see: Wikipedia reference for finding CidrNetmask
- Values based on the example `${VcnCidrBlock}` shown above:
  - `${VcnCidrNetwork}`: 10.0.0.0
  - `${VcnCidrNetmask}`: 255.255.0.0

Config Template Parameter Summary

Each region has multiple Oracle IPSec headends. The template below will allow setting up multiple tunnels on your CPE, each to a corresponding headend. In the table below, "User" is you/your company.

Parameter	Source	Example Value
`${ipAddress1}`	Console/API	129.146.12.52
`${sharedSecret1}`	Console/API	(long string)
`${ipAddress2}`	Console/API	129.146.13.52
`${sharedSecret2}`	Console/API	(long string)
`${cpePublicInterface}`	User	eth1
`${VcnCidrNetwork}`	User	10.0.0.0
`${VcnCidrNetmask}`	User	255.255.0.0

ISAKMP Policy Options

Parameter	Recommended Value
ISAKMP protocol version	Version 1
Exchange type	Main mode
Authentication method	Pre-shared keys
Encryption	AES-256-cbc
Authentication algorithm	SHA-384
Diffie-Hellman Group	Group 5
IKE session key lifetime	28800 seconds (8 hours)

IPSec Policy Options

Parameter	Recommended Value
IPSec protocol	ESP, tunnel-mode
Encryption	AES-256-cbc
Authentication algorithm	HMAC-SHA1-96
Diffie-Hellman Group	Group 5
Perfect Forward Secrecy	Enabled
IPSec session key lifetime	3600 seconds (1 hour)

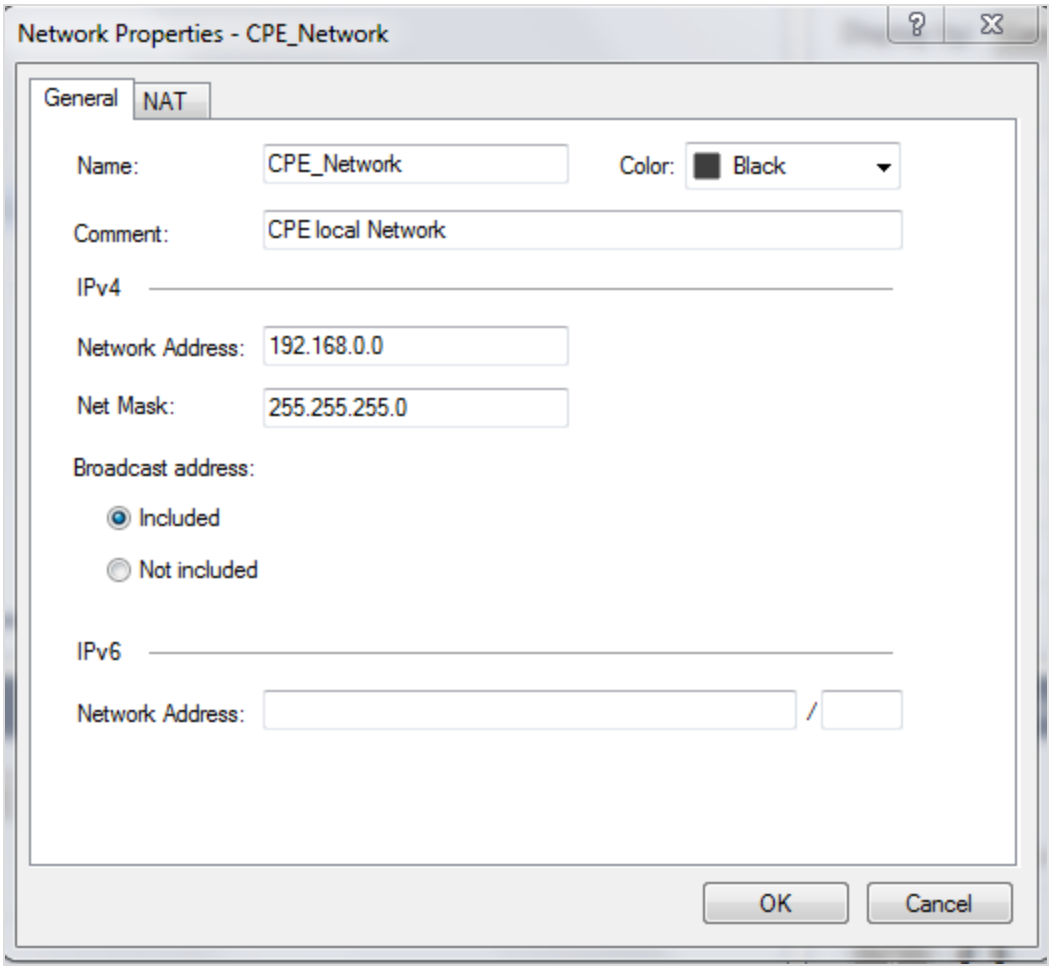
Security Parameter Index

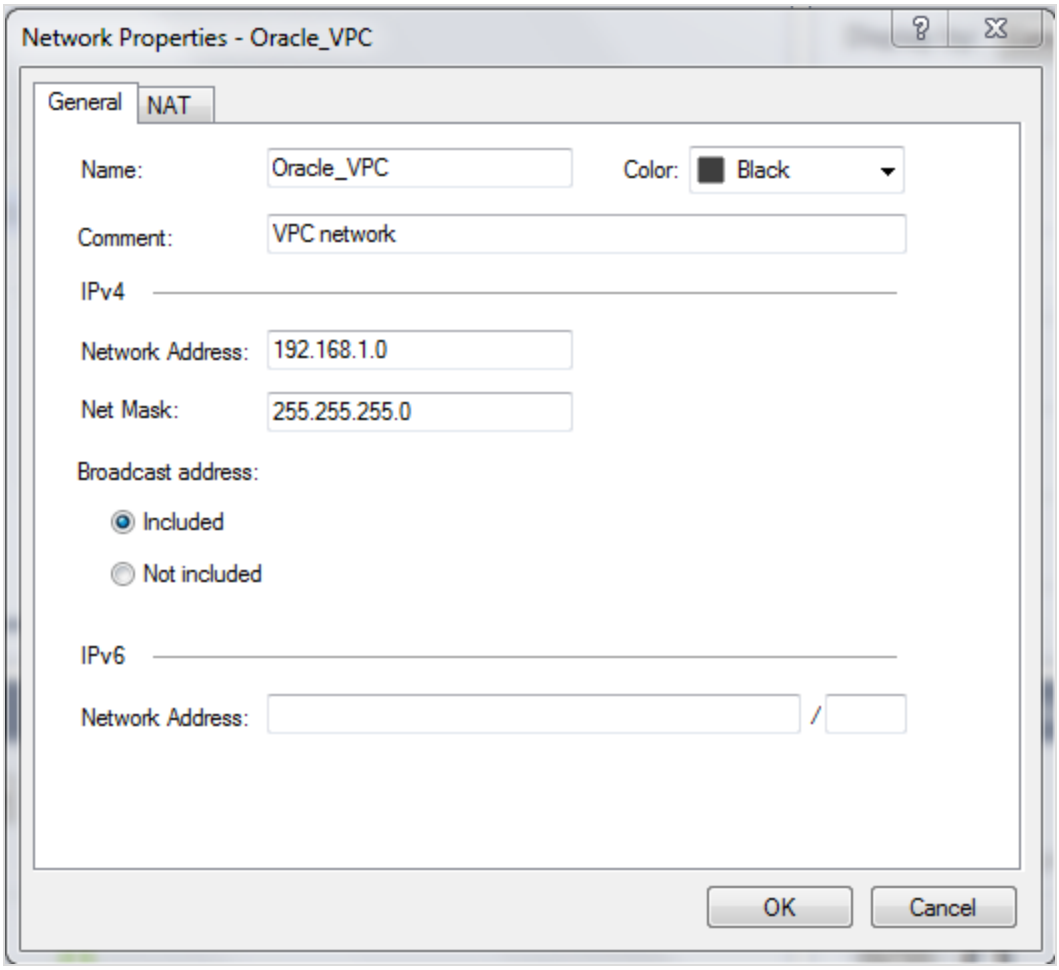
The values for the Security Parameter Index (SPI) depend on whether your CPE supports route-based tunnels or policy-based tunnels. For more information about the correct SPI values to use, see Route-Based Versus Policy-Based IPSec.

Domain-Based Tunnel Configuration

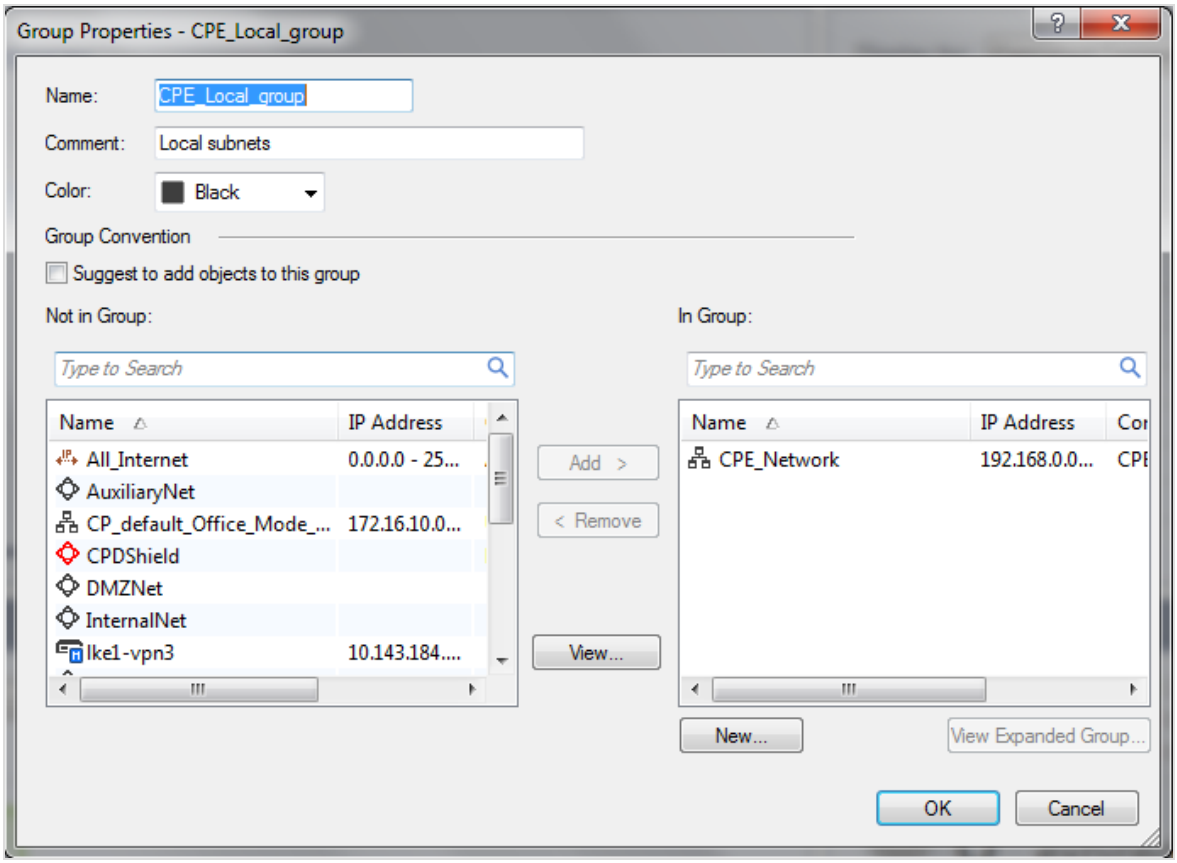
CHECK POINT SMARTDASHBOARD CONFIGURATION

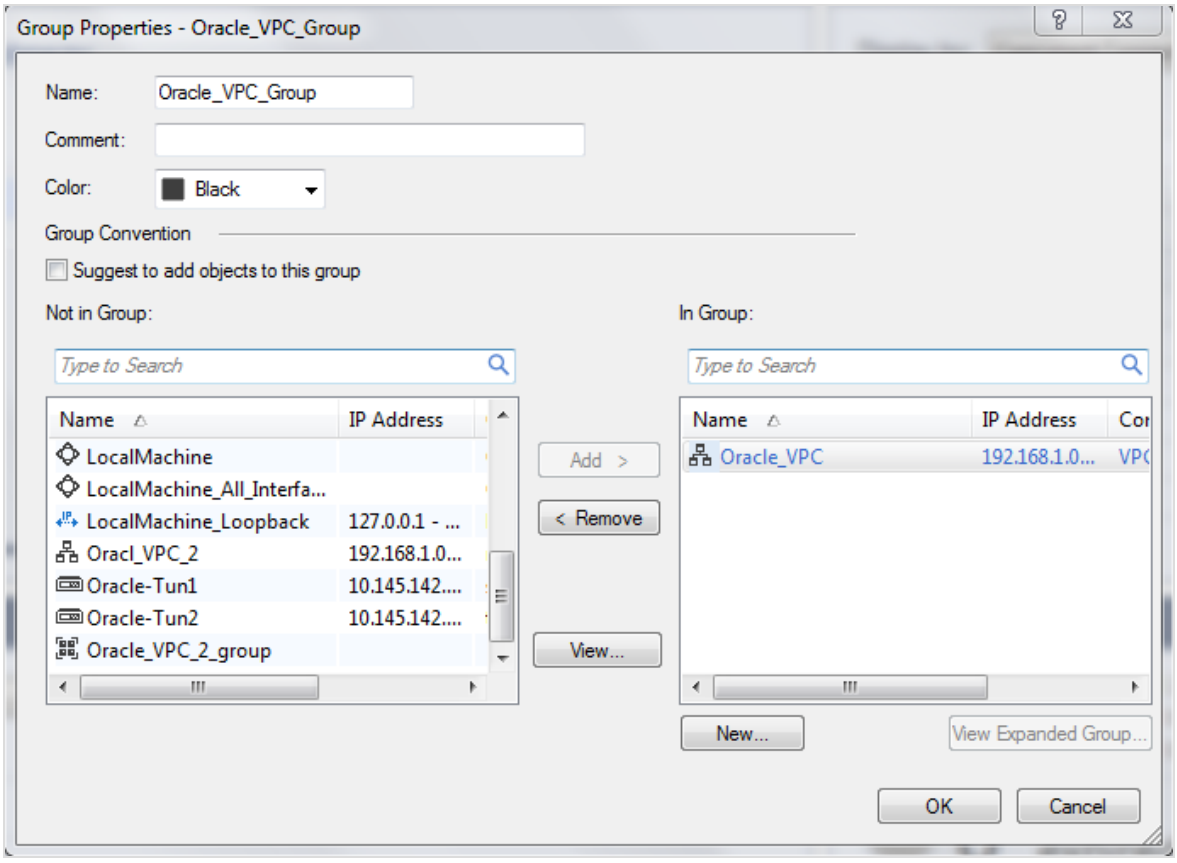
- In SmartDashboard, under the "Networks" section, create new networks to represent the subnets inside your local internal network and the subnets in your Oracle Cloud Infrastructure virtual cloud network (VCN).


Network Properties - CPE_Network
General
NAT
Name:
CPE_Network
Color:
Black
Comment:
CPE local Network
IPv4
Network Address:
192.168.0.0
Net Mask:
255.255.255.0
Broadcast address:
Included
Not included
IPv6
Network Address:
/
OK
Cancel

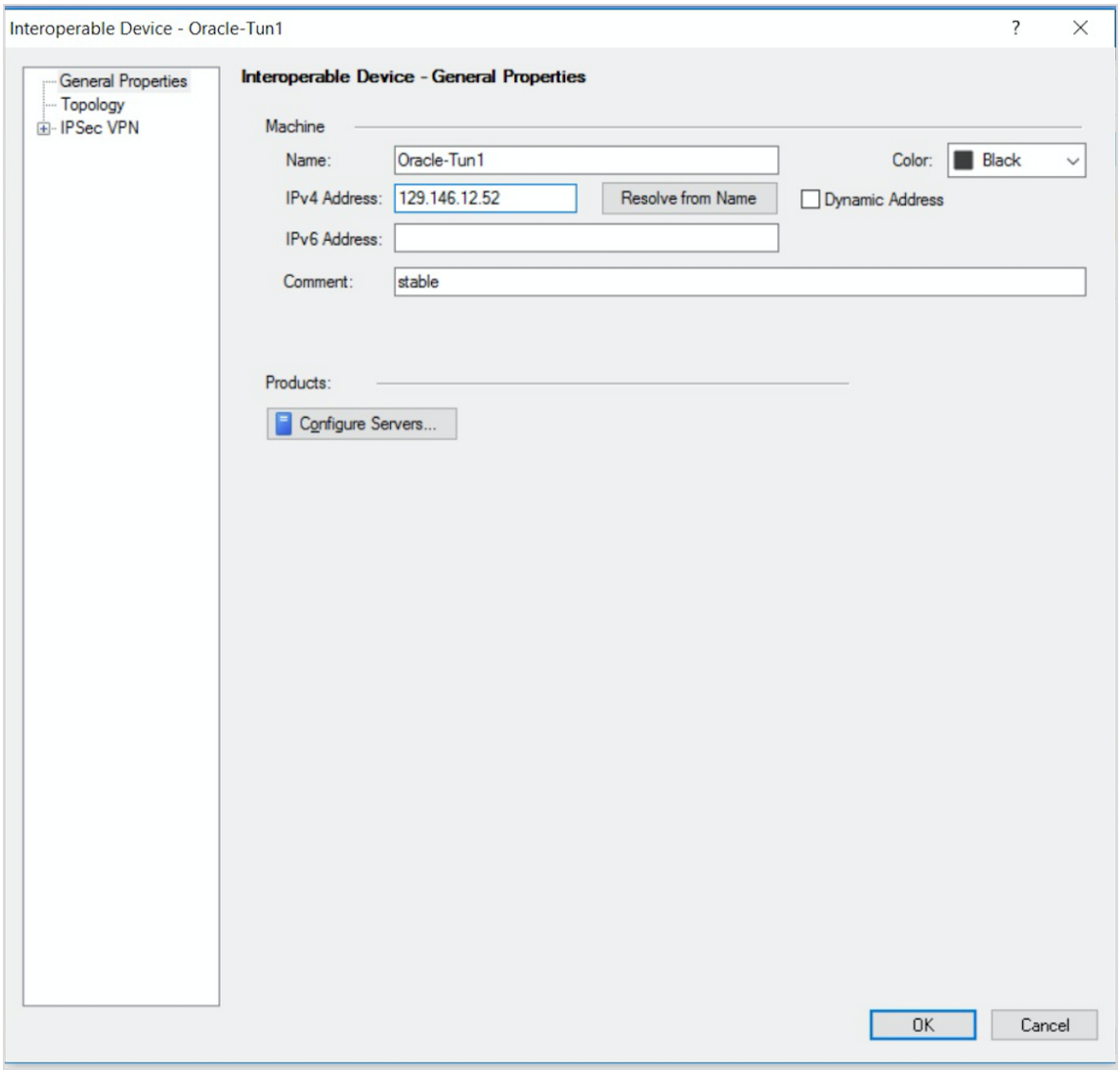

Network Properties - Oracle_VPC
General
NAT
Name:
Oracle_VPC
Color:
Black
Comment:
VPC network
IPv4
Network Address:
192.168.1.0
Net Mask:
255.255.255.0
Broadcast address:
Included
Not included
IPv6
Network Address:
OK
Cancel

- In SmartDashboard, under the groups section, create groups (Simple groups) to represent your and your peer's VPN domain. Groups are required when multiple subnets need to be reachable over the VPN tunnel. In this example you're adding only one network object to the group, but you can add multiple according to your network's topology. Check Point recommends not adding a group as a part of another group because performance can be impacted.

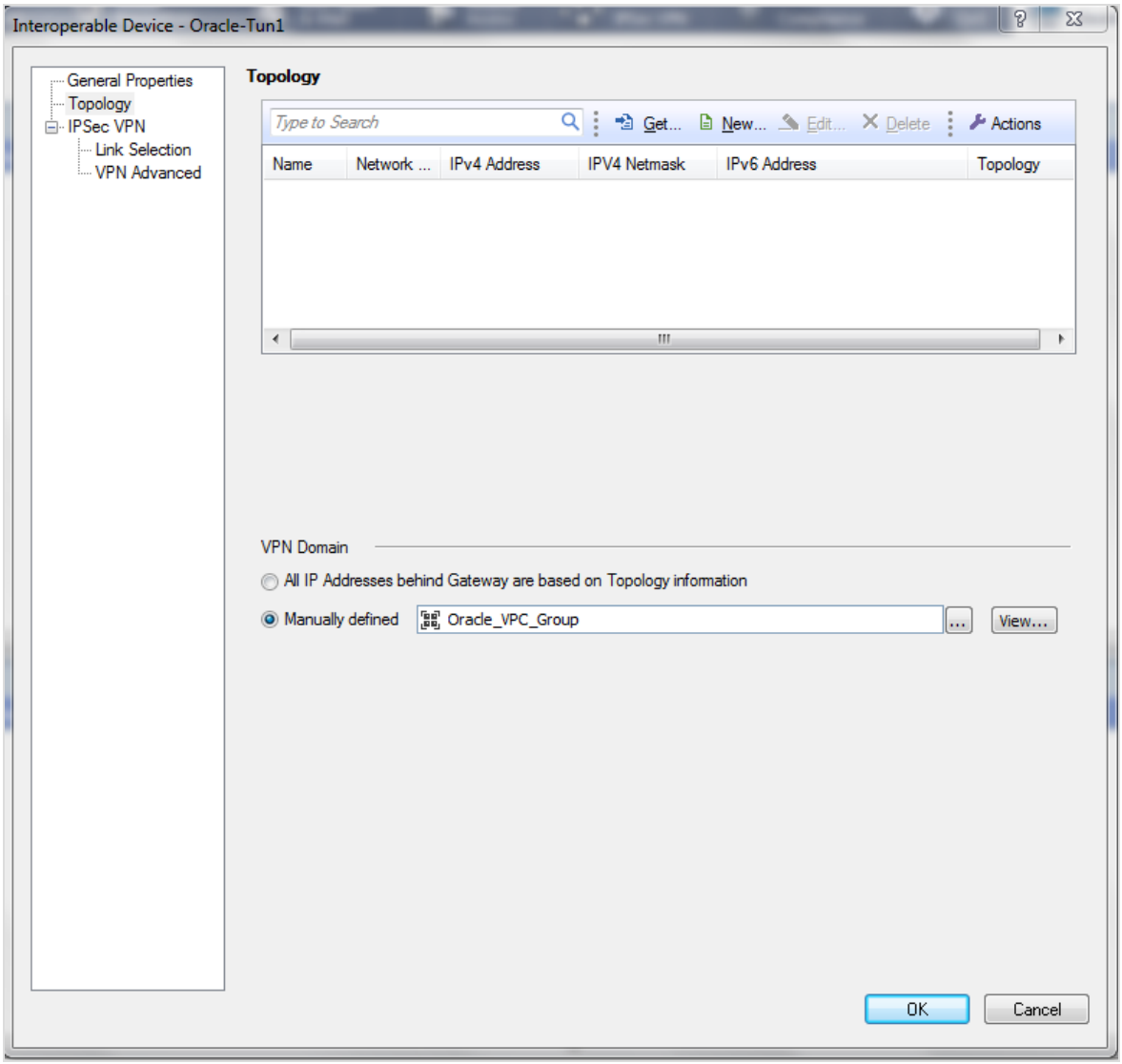




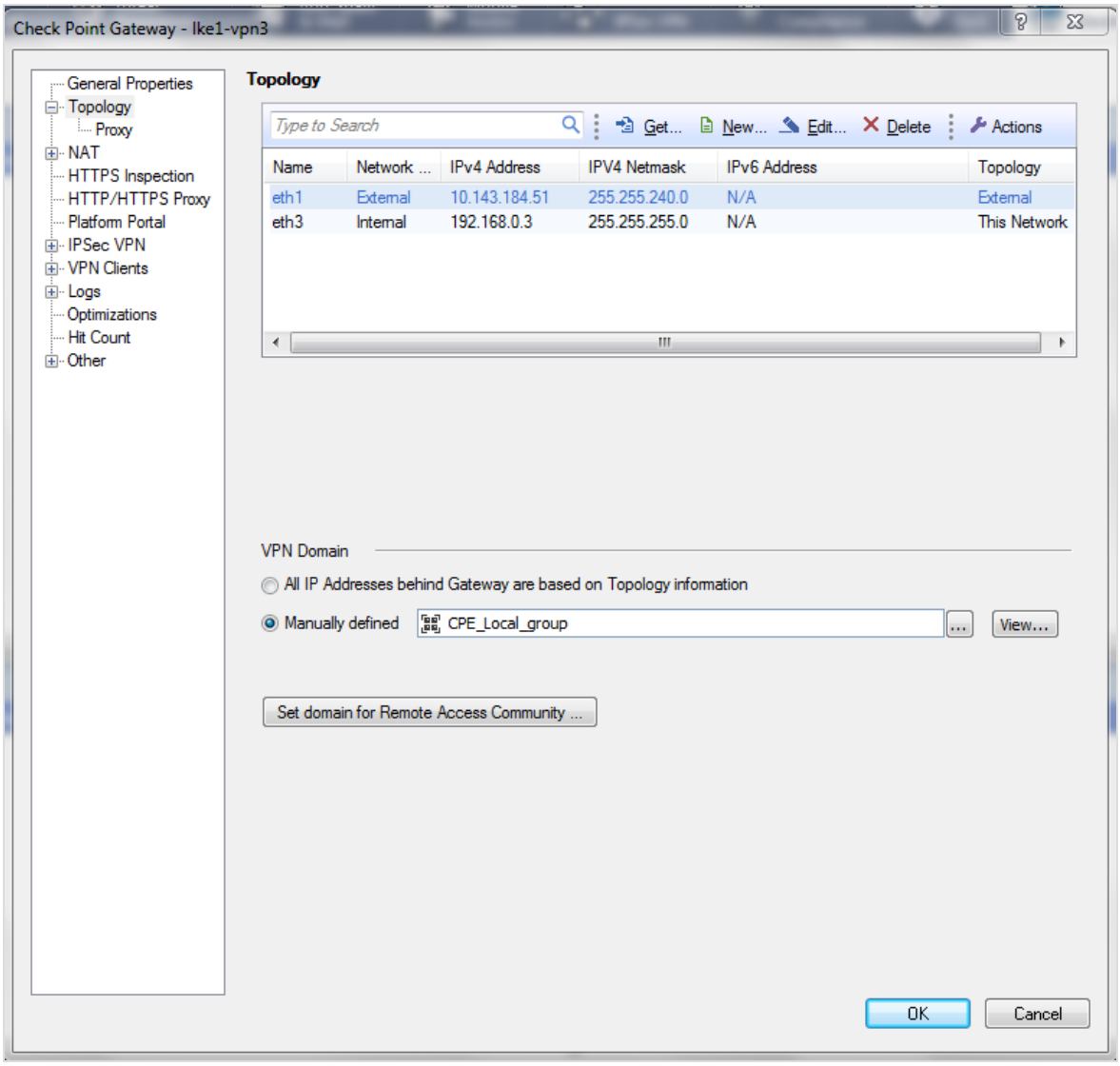
- In SmartDashboard, create a new Network Object, "Interoperable Device...", one per tunnel.
    - Name: Unique name per tunnel
    - IPv4 Address: ${ipAddress#}


Interoperable Device - Oracle-Tun1
General Properties
Topology
IPSec VPN
Interoperable Device - General Properties
Machine
Name:
Oracle-Tun1
Color:
Black
IPv4 Address:
129.146.12.52
Resolve from Name
Dynamic Address
IPv6 Address:
Comment:
stable
Products:
Configure Servers...
OK
Cancel

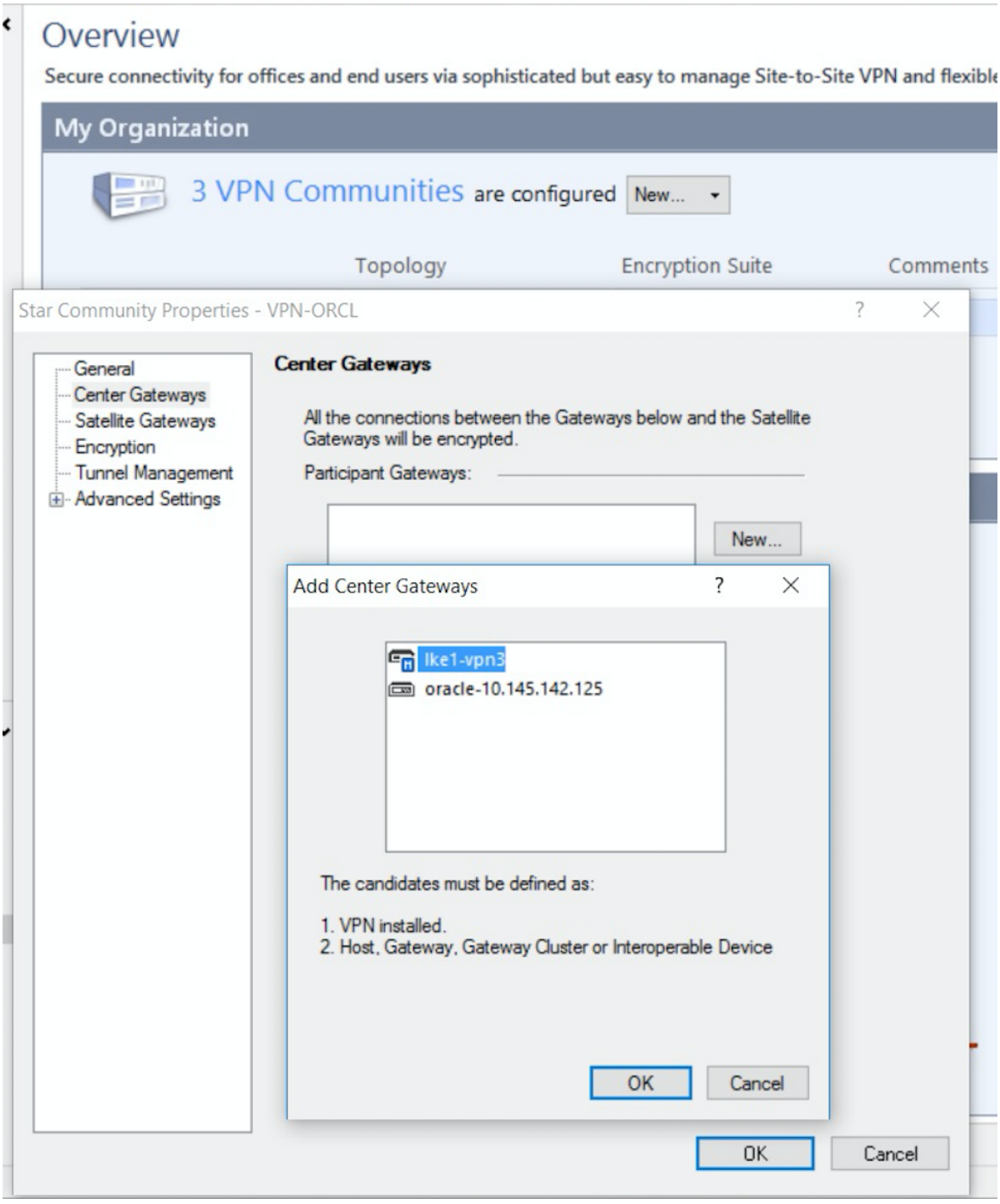
In this device's Topology tab, under the VPN domain, select "manually defined", and then select the group that you created in the previous steps that represents the remote side (Oracle/VCN side) domain.



- Open your local device and go to the topology tab. Under the VPN domain, select "manually defined", and then select the group that you created in the previous steps that represents your internal domain.



- Open IPsec VPN tab "Communities" and create new "Star Community".
    - Add your gateway or cluster as the Center Gateway.
    - Add new Interoperable Devices as Satellite Gateways.

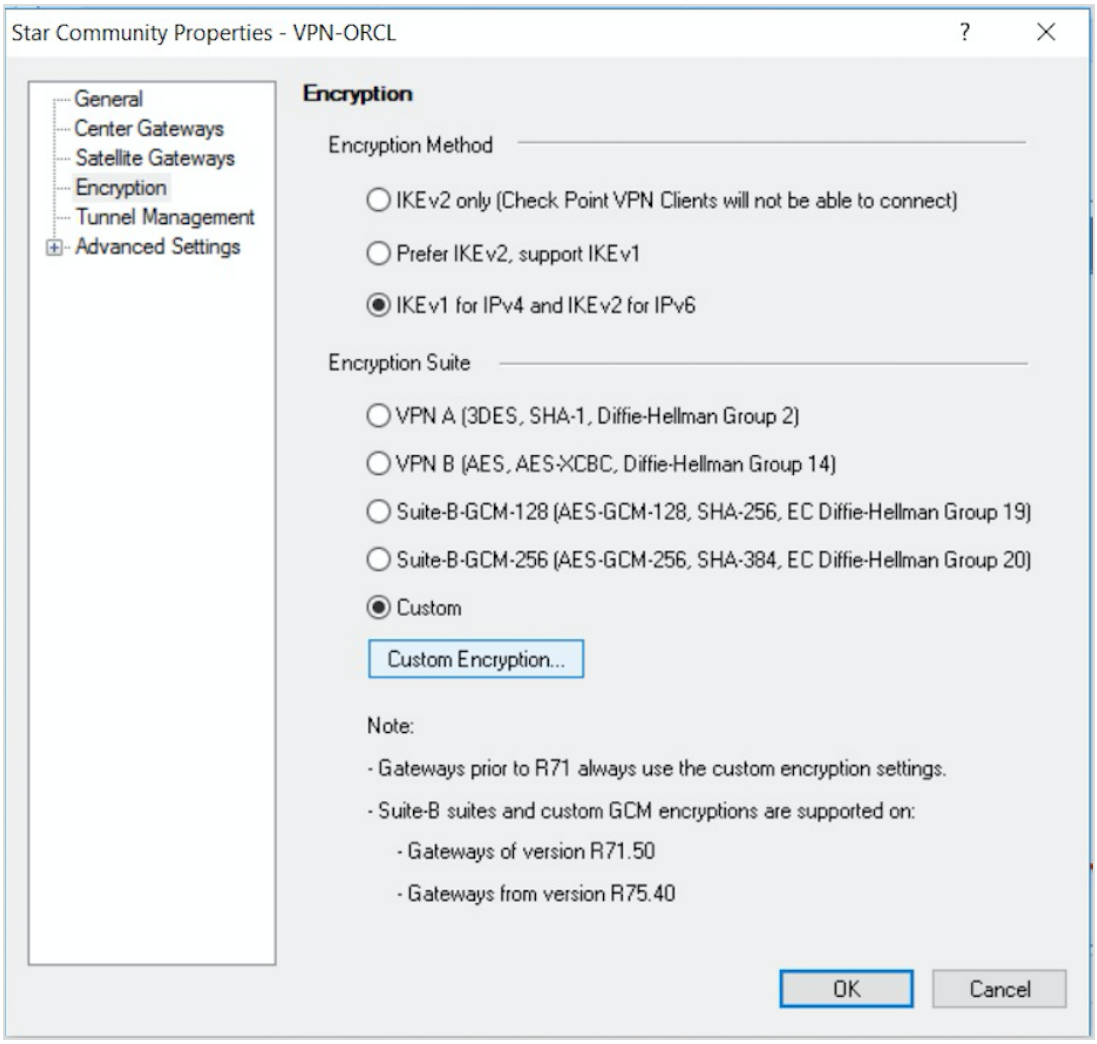

Overview
Secure connectivity for offices and end users via sophisticated but easy to manage Site-to-Site VPN and flexible
My Organization
3 VPN Communities are configured
New...
Topology
Encryption Suite
Comments
Star Community Properties - VPN-ORCL
General
Center Gateways
Satellite Gateways
Encryption
Tunnel Management
Advanced Settings
Center Gateways
All the connections between the Gateways below and the Satellite Gateways will be encrypted.
Participant Gateways:
New...
Add Center Gateways
ike1-vpn3
oracle-10.145.142.125
The candidates must be defined as:
1. VPN installed.
2. Host, Gateway, Gateway Cluster or Interoperable Device
OK
Cancel
OK
Cancel

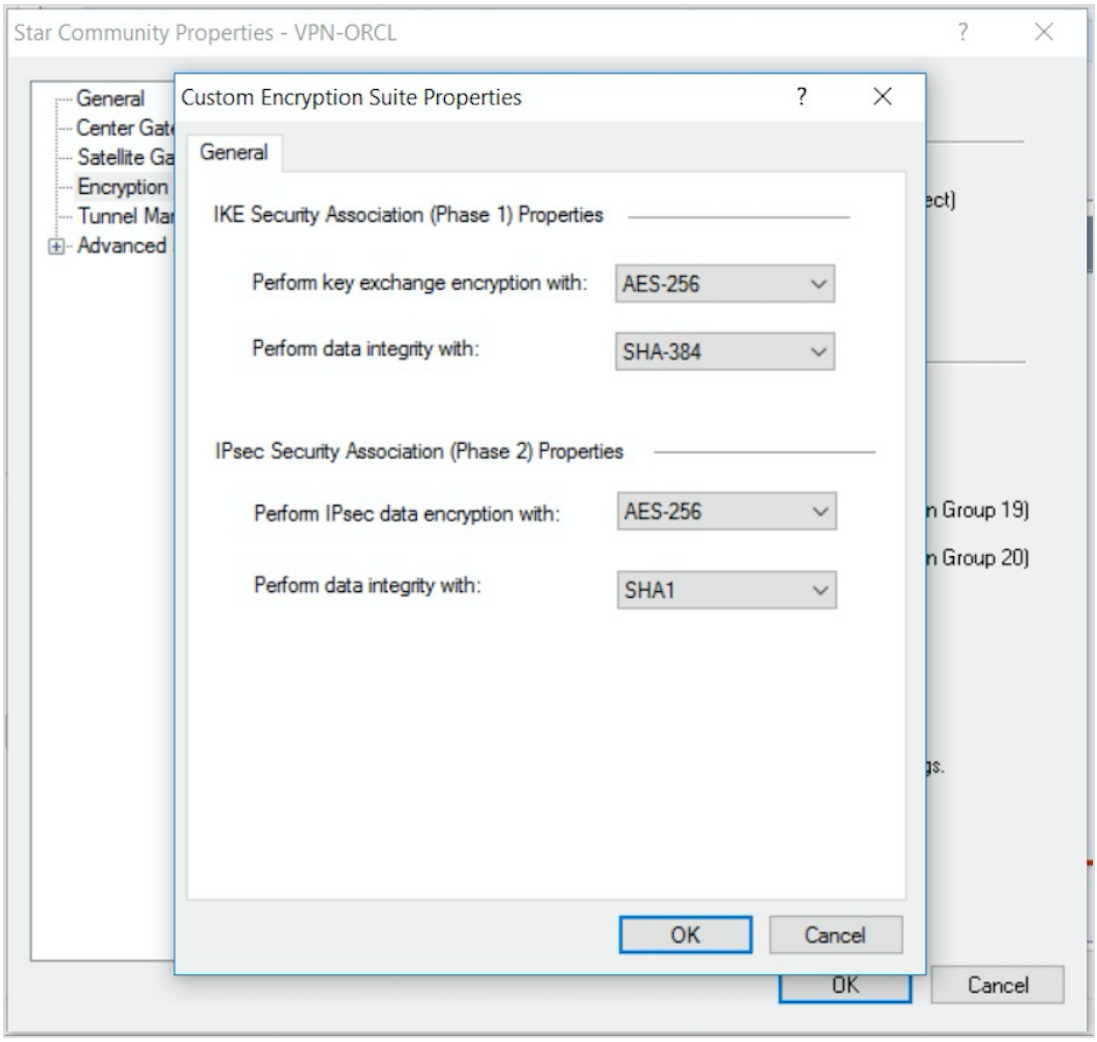
- Open the "Encryption" tab > Encryption Method, select "IKEv1" and under Encryption Suite select "Custom". Select the following parameters for custom encryption suite:
  - ISAKMP Phase 1: IKE SA
    - AES-256
    - SHA-384
  - ISAKMP Phase 2: IPSec SA
    - AES-256
    - SHA1



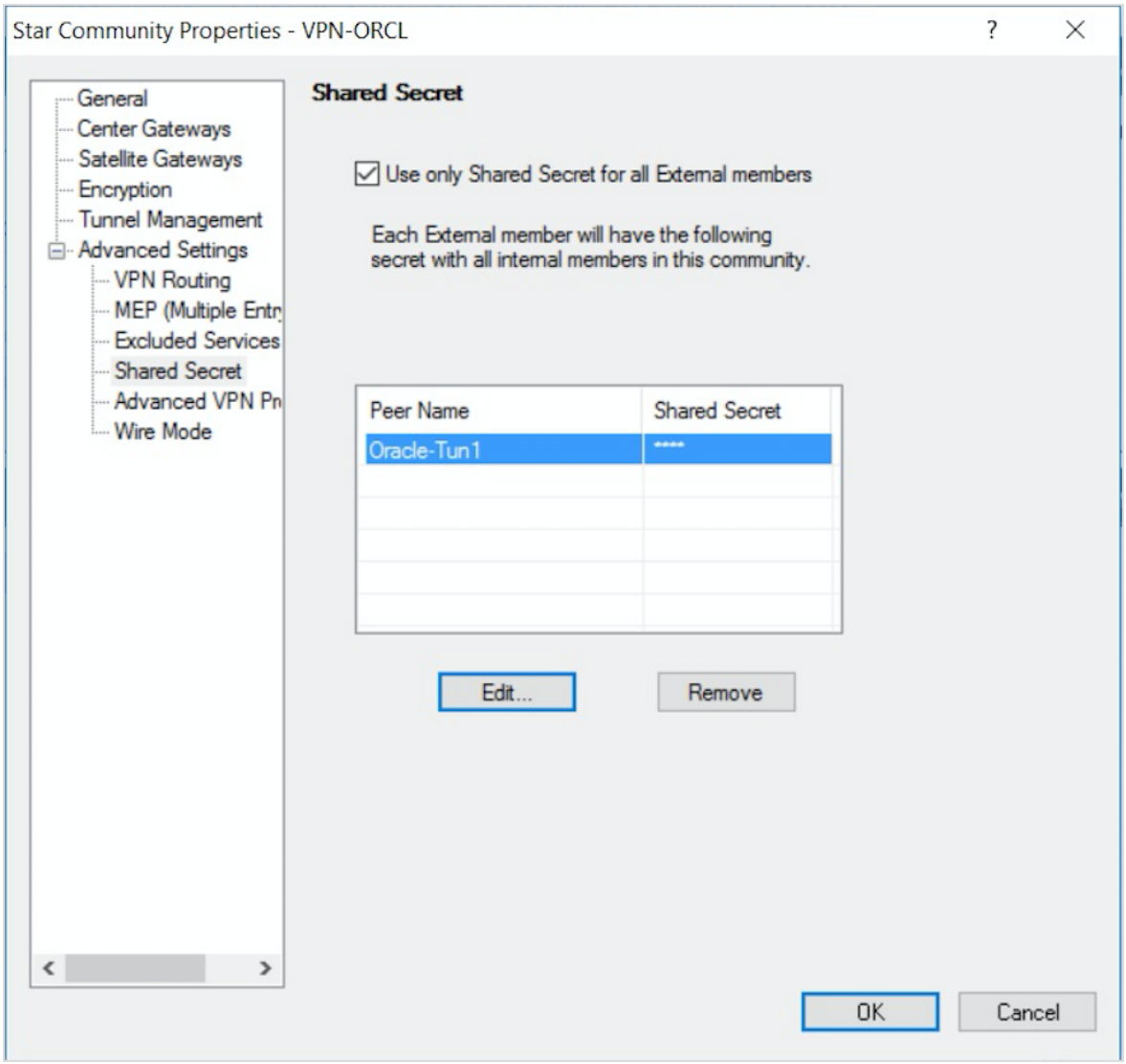
Note

Use of SHA-1 is recommended because there are issues setting up the IPSec tunnels if you're using older versions of Check Point software (before R77.30) with SHA-384. For more information, see VPN tunnel can not be established / no traffic passes when SHA-384 is configured for data integrity.


Star Community Properties - VPN-ORCL
General
Center Gateways
Satellite Gateways
Encryption
Tunnel Management
Advanced Settings
Encryption
Encryption Method
IKEv2 only (Check Point VPN Clients will not be able to connect)
Prefer IKEv2, support IKEv1
IKEv1 for IPv4 and IKEv2 for IPv6
Encryption Suite
VPN A (3DES, SHA-1, Diffie-Hellman Group 2)
VPN B (AES, AES-XCBC, Diffie-Hellman Group 14)
Suite-B-GCM-128 (AES-GCM-128, SHA-256, EC Diffie-Hellman Group 19)
Suite-B-GCM-256 (AES-GCM-256, SHA-384, EC Diffie-Hellman Group 20)
Custom
Custom Encryption...
Note:
- Gateways prior to R71 always use the custom encryption settings.
- Suite-B suites and custom GCM encryptions are supported on:
- Gateways of version R71.50
- Gateways from version R75.40
OK
Cancel

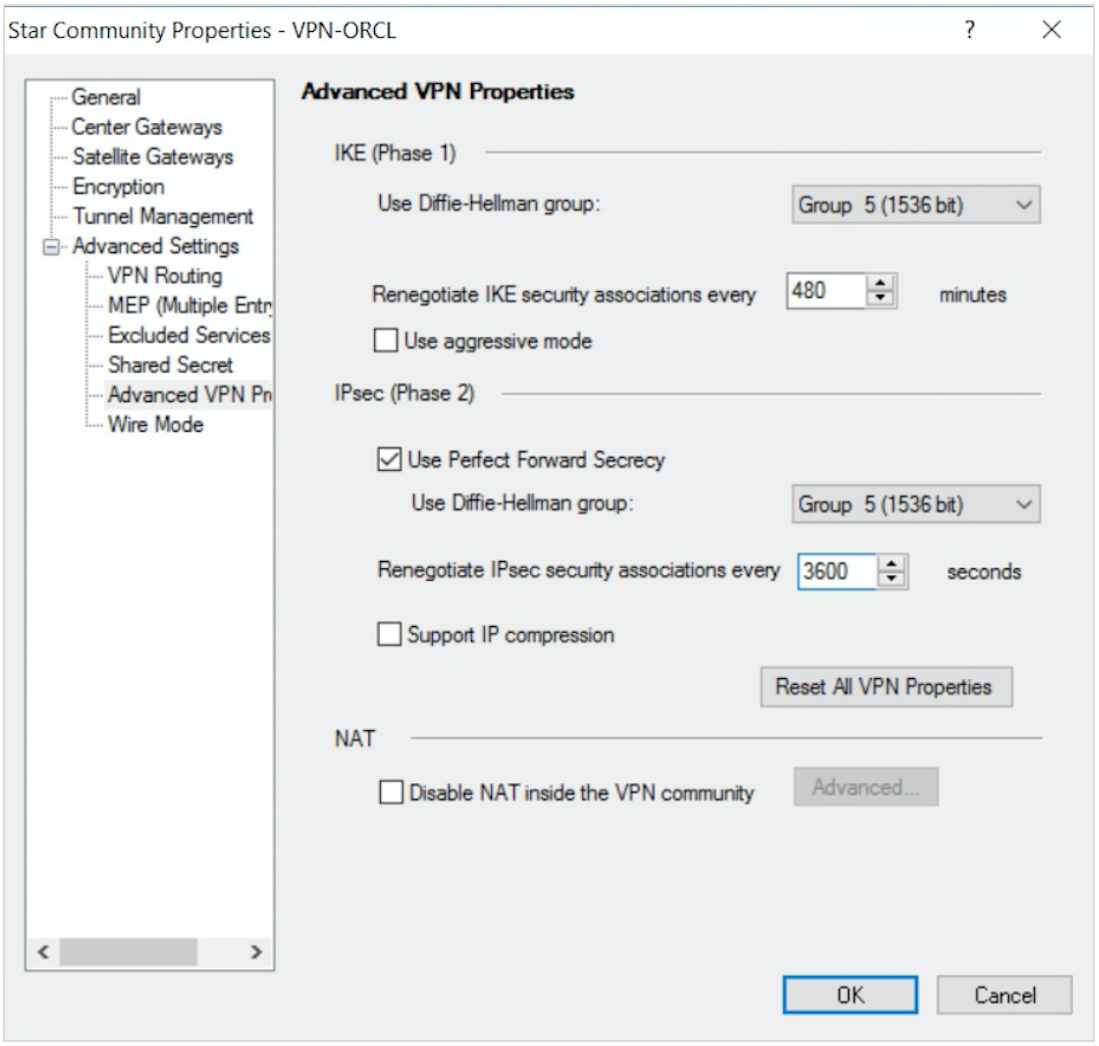


- Under "Advanced Settings" > "Shared Secret", configure pre-shared keys.
  - For each Peer, there is a unique ${sharedSecret#}

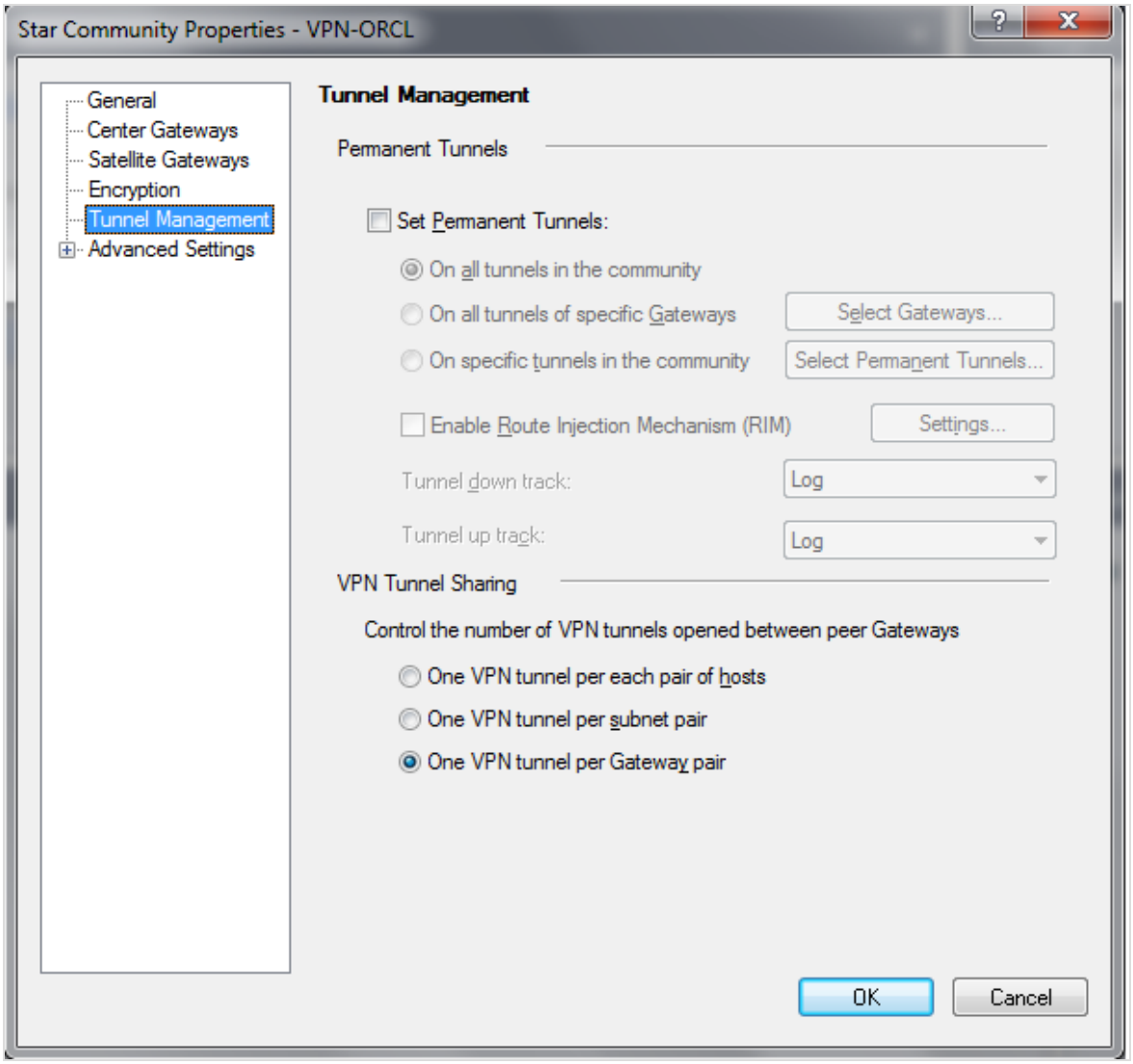


- Under "Advanced Settings" > "Advanced VPN Properties":

- IKE (Phase 1):
  - Use Diffie-Hellman group: Group 5
  - Renegotiate IKE security associations every: 480 minutes
  - Uncheck "Use aggressive mode"
- IPsec (Phase 2):
  - Check "Use Perfect Forward Secrecy"
  - Use Diffie-Hellman group: Group 5
  - Renegotiate IPsec security associations every: 3600 seconds


Star Community Properties - VPN-ORCL
?
General
Center Gateways
Satellite Gateways
Encryption
Tunnel Management
Advanced Settings
VPN Routing
MEP (Multiple Entry
Excluded Services
Shared Secret
Advanced VPN Pr
Wire Mode
Advanced VPN Properties
IKE (Phase 1)
Use Diffie-Hellman group:
Group 5 (1536 bit)
Renegotiate IKE security associations every
480
minutes
Use aggressive mode
IPsec (Phase 2)
Use Perfect Forward Secrecy
Use Diffie-Hellman group:
Group 5 (1536 bit)
Renegotiate IPsec security associations every
3600
seconds
Support IP compression
Reset All VPN Properties
NAT
Disable NAT inside the VPN community
Advanced...
OK
Cancel

- Under "Tunnel Management", under "VPN Tunnel Sharing", select "One VPN tunnel per Gateway pair".





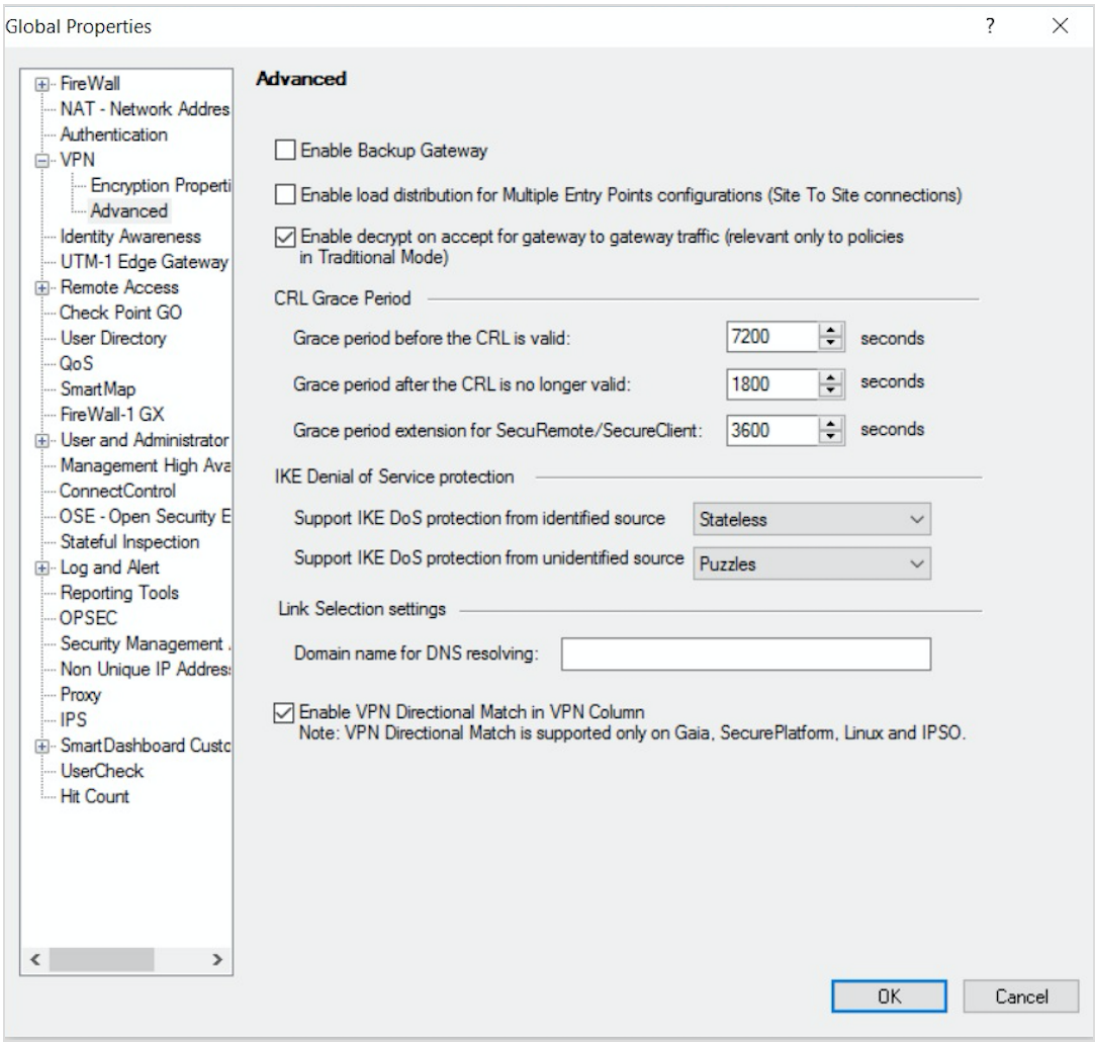
Important

You can select Tunnel Management parameters under both "VPN community" and "Security Gateway Object". **Make sure you select "One VPN tunnel per Gateway pair" under both of these sections, or else you may have connectivity issues due to conflicting settings.**

More details: If there's a conflict between the tunnel properties of a VPN community and a Security Gateway object that is a member of that same community, the stricter setting is followed. For example, a Security Gateway set to "One VPN Tunnel per each pair of hosts" and a community set to "One VPN Tunnel per Gateway pair" would follow "One VPN Tunnel per each pair of hosts".

- Create Firewall Rules: Open Global Properties > VPN > Advanced.

- Check "Enable VPN Directional Match"



- For any firewall rule that matches VPN traffic, add match rules for each direction.



- Install firewall policy



Note

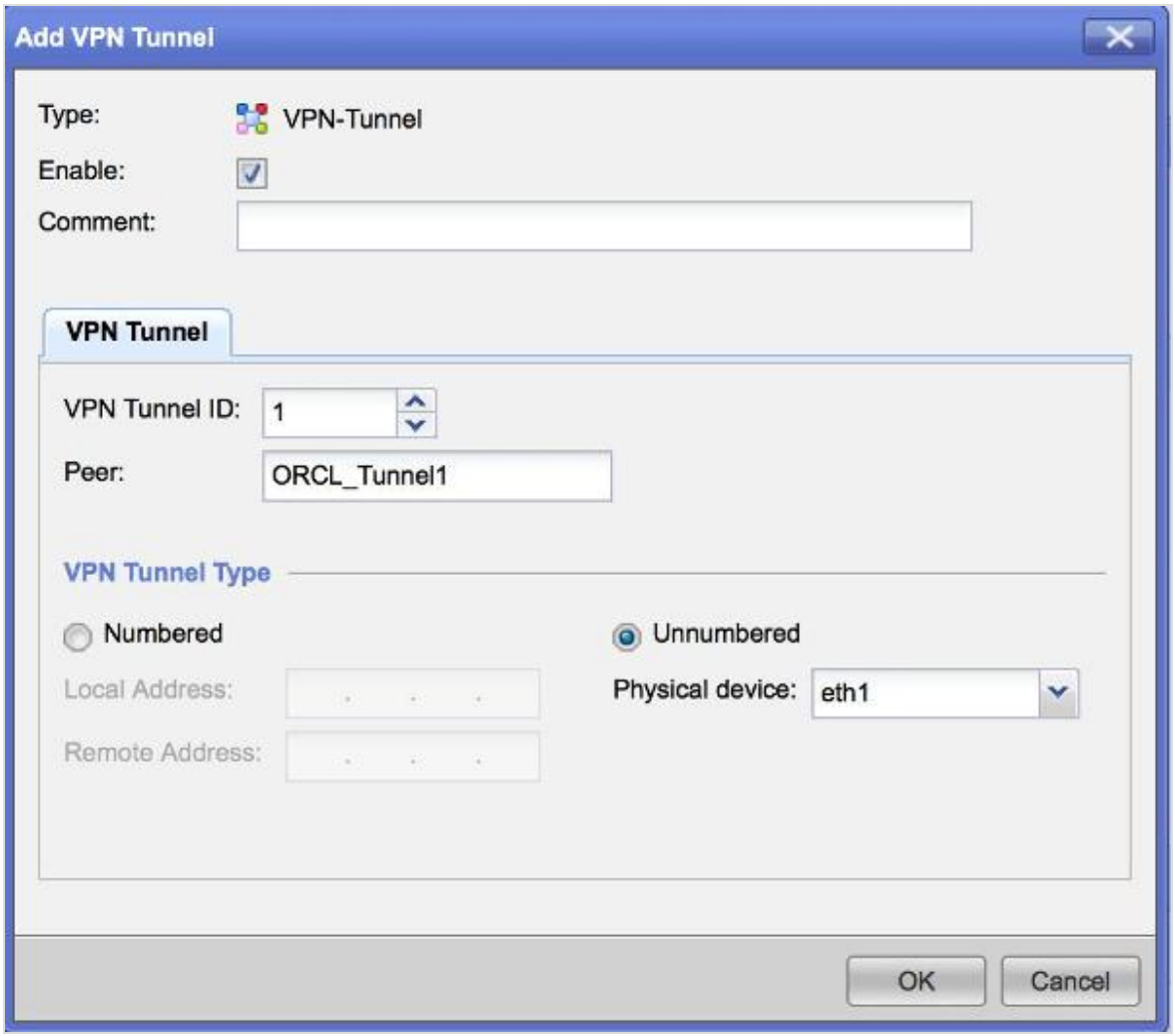
Redundant tunnel configuration is not recommended because the tunnel failover will not work due to absence of an IP address (numbered VTIs or BGP IPSec) that can be used to ping the peer for DPD and failover. If you need redundant tunnels, please contact your Oracle representative.

VPN Tunnel Interface (VTI) Configuration

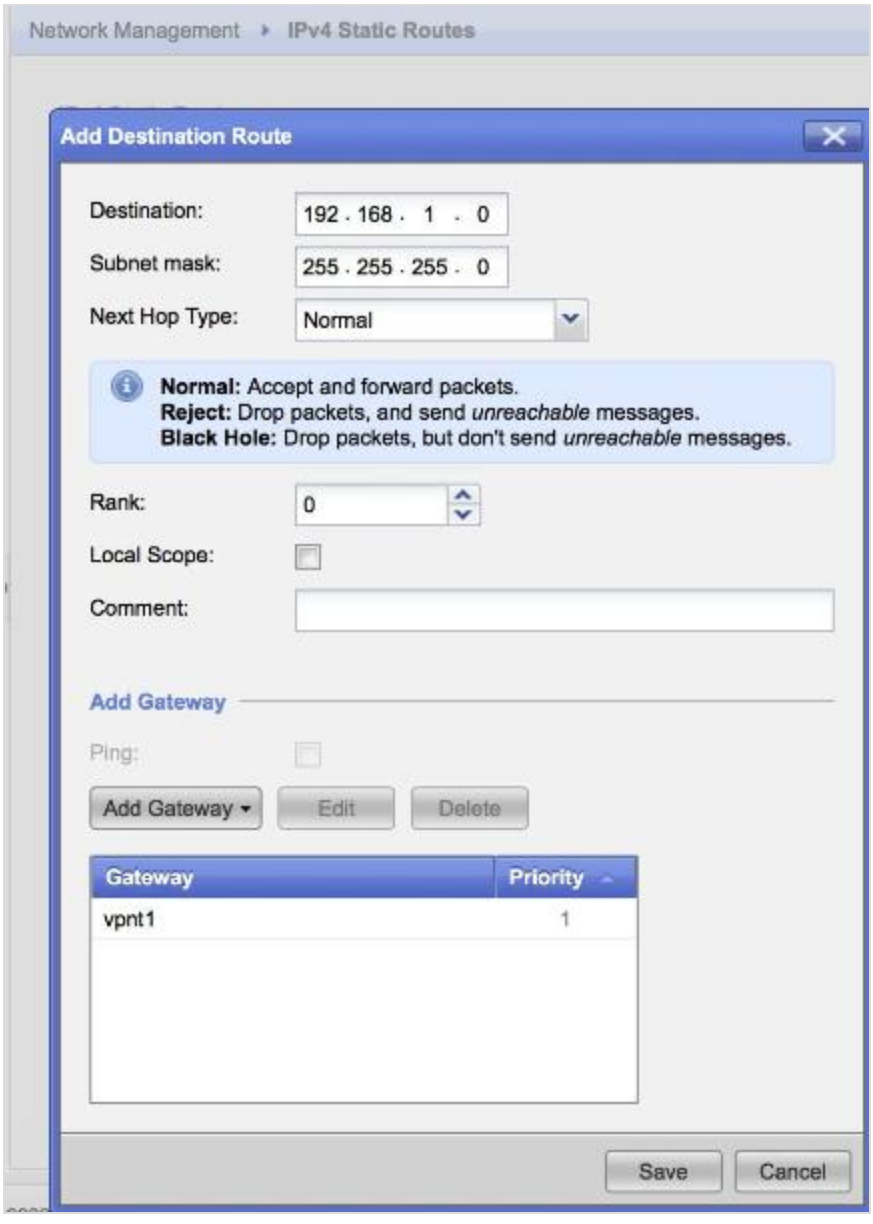
Check Point Gaia Portal Configuration

- Under "Network Interfaces" tab, create one new "VPN Tunnel" interface per IPSec connection tunnel.
    - VPN Tunnel ID: Select a unique number
    - Peer: Unique name for tunnel
    - VPN Tunnel Type: Unnumbered
    - Physical Device: ${cpePublicInterface}


Network Management
Network Interfaces
Interfaces
Add
Edit
Delete
Alias
VLAN
Bond
Bridge
Loopback
VPN Tunnel
6in4 Tunnel
PPPoE
Ethernet
Ethernet
Ethernet
Ethernet
Ethernet
Ethernet
Loopback



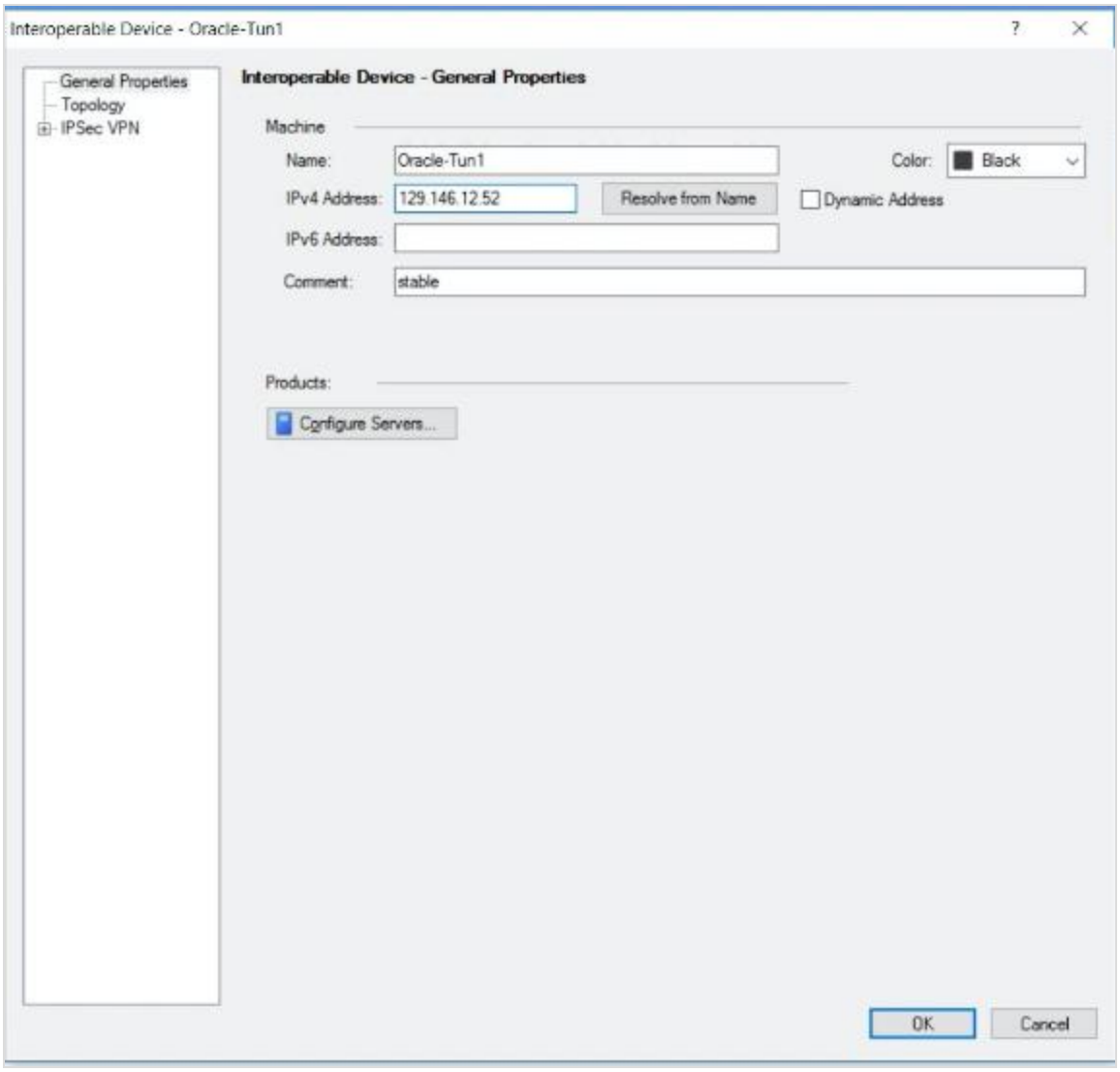
- Under the "IPv4 Static Routes" tab, define VPN static routes, one per tunnel.
  - Destination: ${VcnCidrNetwork}
  - Subnet Mask: ${VcnCidrNetmask}
  - Gateway: Interface - One per VPN tunnel



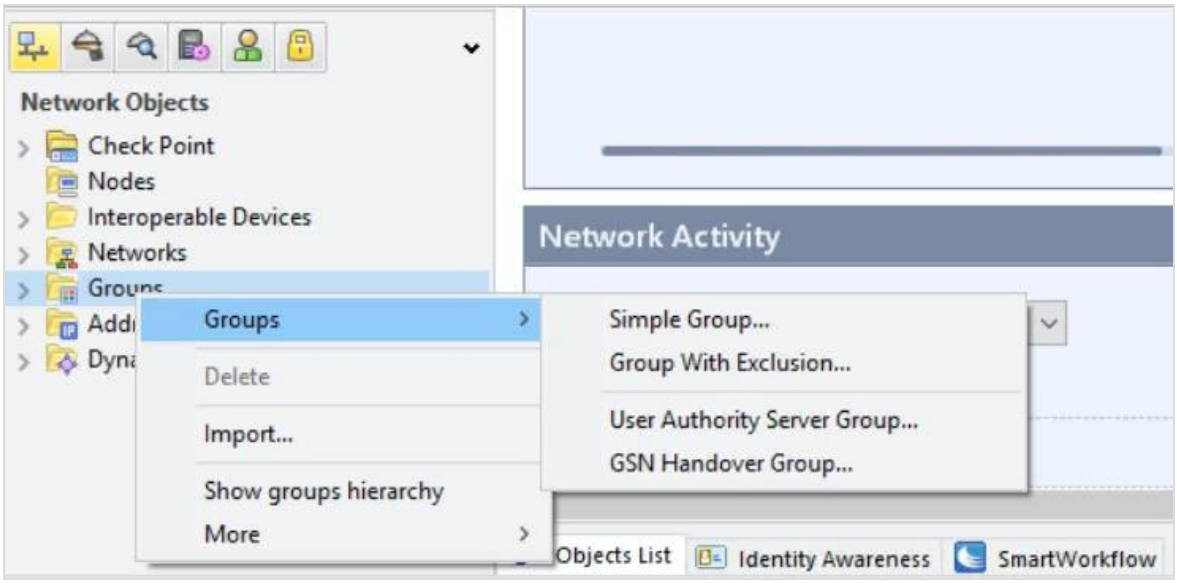
- In SmartDashboard, create a new Network Object, "Interoperable Device...", one per

tunnel.

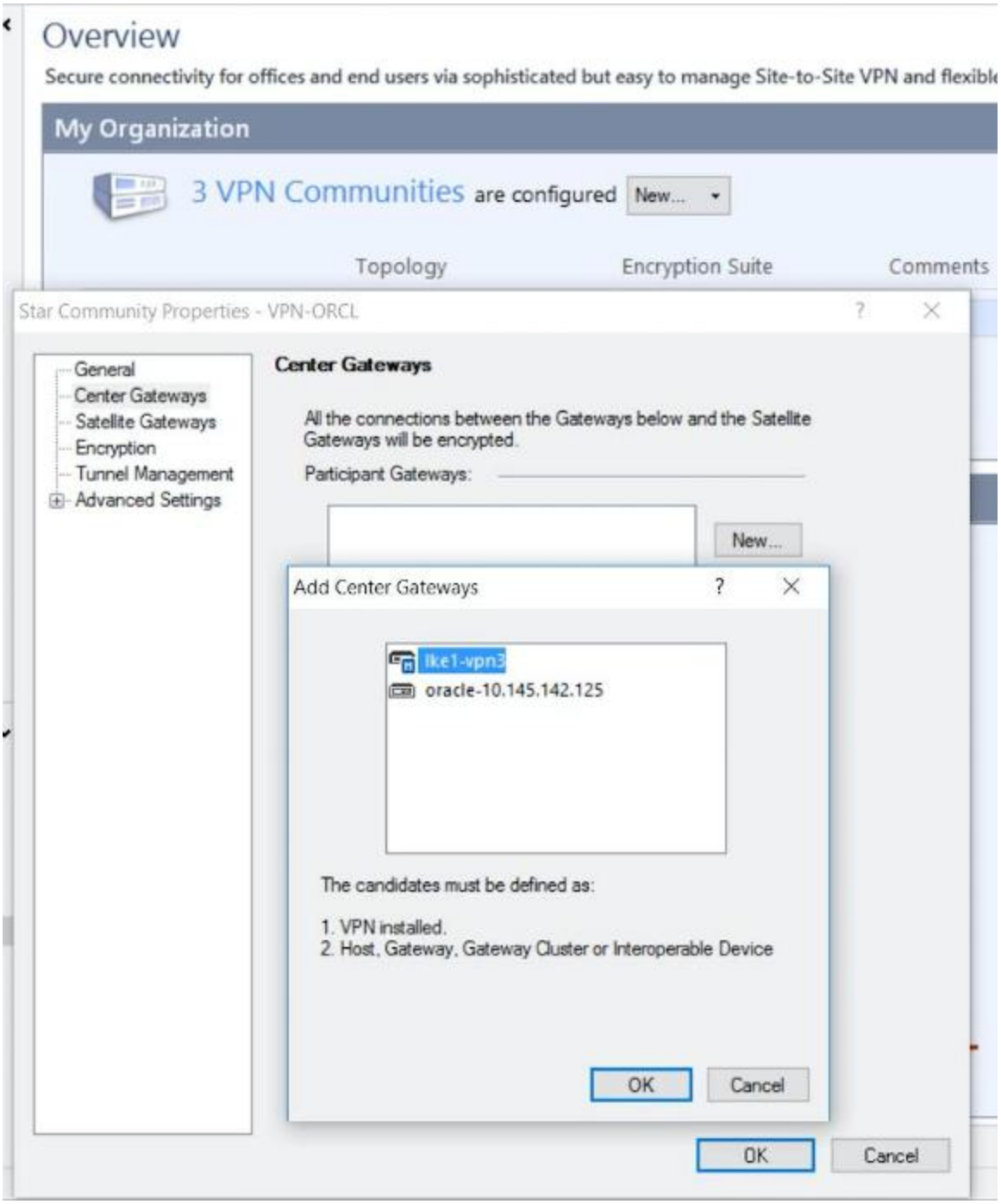
- Name: Unique name per tunnel
- IPv4 Address: ${ipAddress#}


Interoperable Device - Oracle-Tun1
General Properties
Topology
IPSec VPN
Interoperable Device - General Properties
Machine
Name:
Oracle-Tun1
Color:
Black
IPv4 Address:
129.146.12.52
Resolve from Name
Dynamic Address
IPv6 Address:
Comment:
stable
Products:
Configure Servers...
OK
Cancel

- In SmartDashboard, create a new group, "Simple Group..."



- Open your gateway/cluster and navigate to the "Topology" tab, choose "Manually defined", and select new Simple Group.
- Open IPsec VPN tab "Communities" and create new "Star Community".
    - Add your gateway or cluster as the Center Gateway.
    - Add new Interoperable Devices as Satellite Gateways.

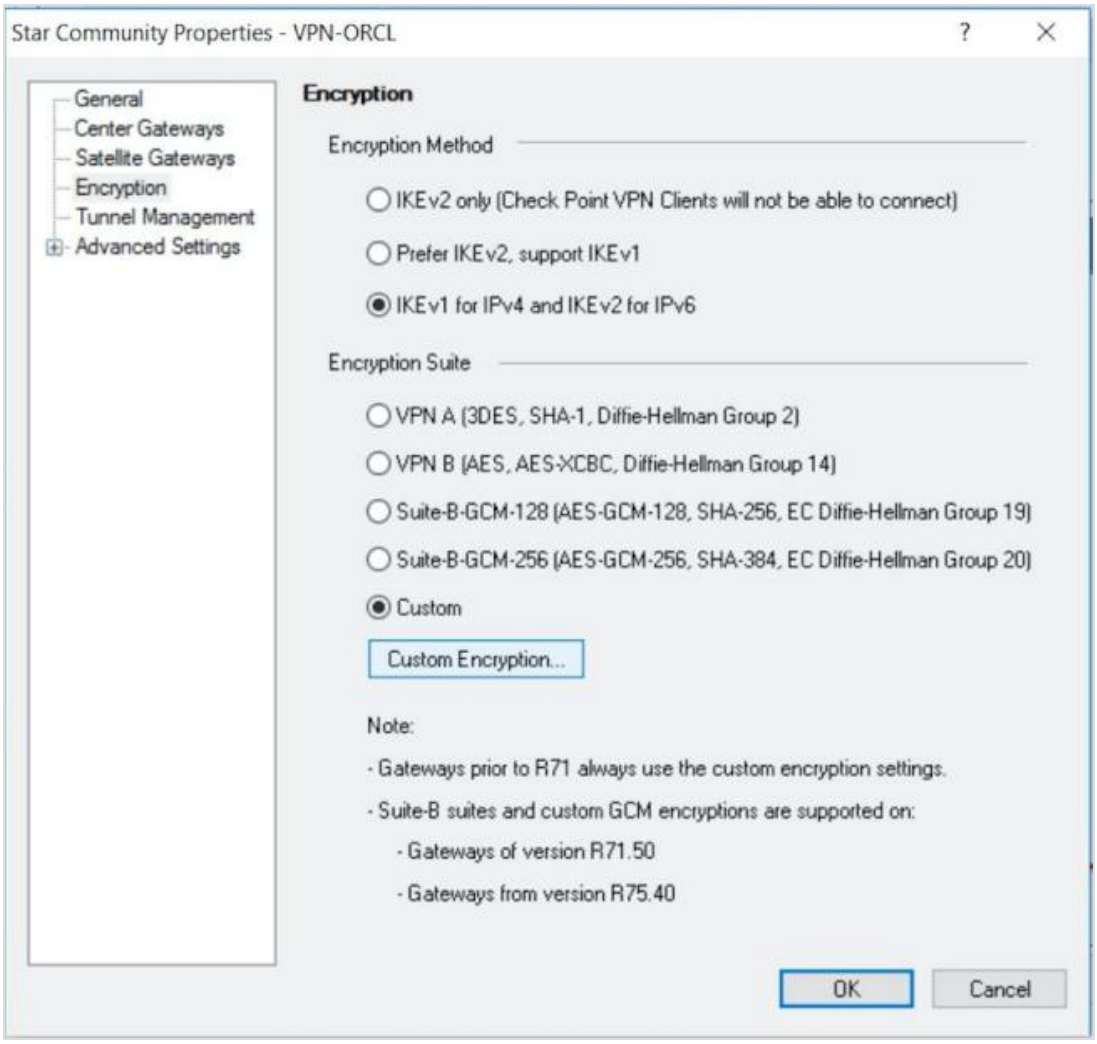

Overview
Secure connectivity for offices and end users via sophisticated but easy to manage Site-to-Site VPN and flexibl
My Organization
3 VPN Communities are configured
New...
Topology
Encryption Suite
Comments
Star Community Properties - VPN-ORCL
General
Center Gateways
Satellite Gateways
Encryption
Tunnel Management
Advanced Settings
Center Gateways
All the connections between the Gateways below and the Satellite Gateways will be encrypted.
Participant Gateways:
New...
Add Center Gateways
ike1-vpn3
oracle-10.145.142.125
The candidates must be defined as:
1. VPN installed.
2. Host, Gateway, Gateway Cluster or Interoperable Device
OK
Cancel
OK
Cancel

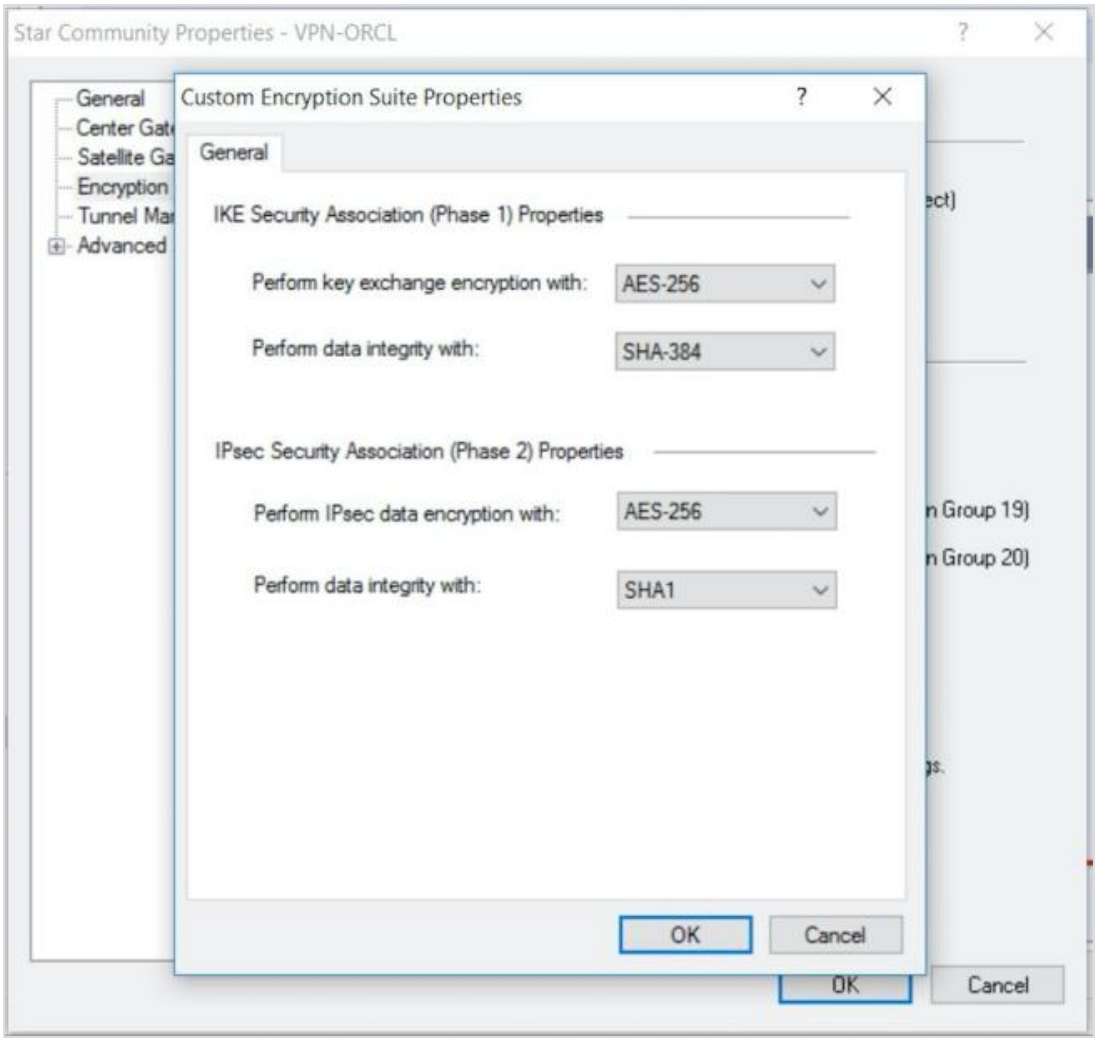
- Open the "Encryption" tab > Encryption Method, select "IKEv1" and under Encryption Suite select "Custom". Select the following parameters for custom encryption suite:
  - ISAKMP Phase 1: IKE SA
    - AES-256
    - SHA-384
  - ISAKMP Phase 2: IPSec SA
    - AES-256
    - SHA1



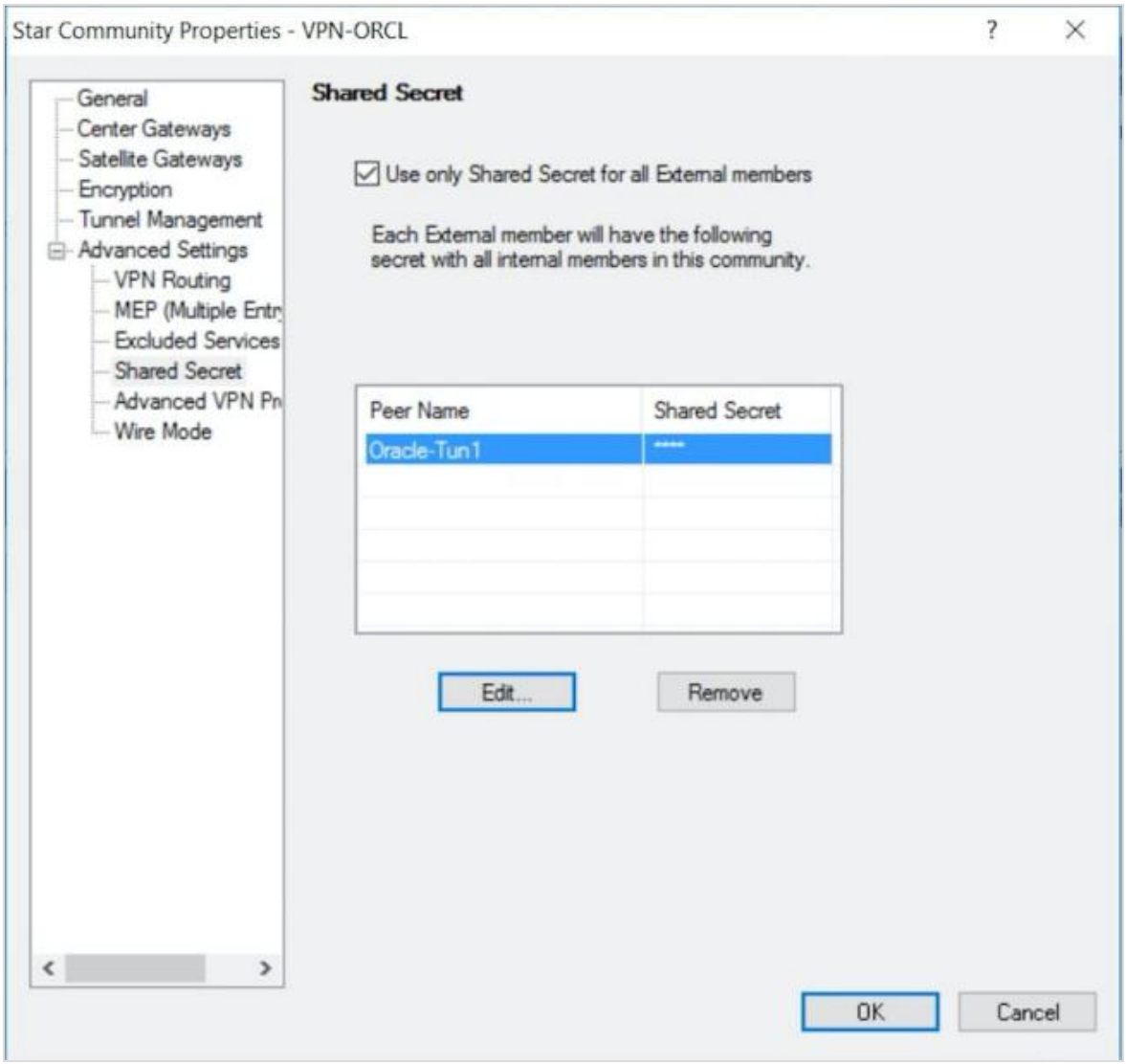
Note

Use of SHA-1 is recommended because there are issues setting up the IPSec tunnels if you're using older versions of Check Point software (before R77.30) with SHA-384. For more information, see VPN tunnel can not be established / no traffic passes when SHA-384 is configured for data integrity.


Star Community Properties - VPN-ORCL
General
Center Gateways
Satellite Gateways
Encryption
Tunnel Management
Advanced Settings
Encryption
Encryption Method
IKEv2 only (Check Point VPN Clients will not be able to connect)
Prefer IKEv2, support IKEv1
IKEv1 for IPv4 and IKEv2 for IPv6
Encryption Suite
VPN A (3DES, SHA-1, Diffie-Hellman Group 2)
VPN B (AES, AES-XCBC, Diffie-Hellman Group 14)
Suite-B-GCM-128 (AES-GCM-128, SHA-256, EC Diffie-Hellman Group 19)
Suite-B-GCM-256 (AES-GCM-256, SHA-384, EC Diffie-Hellman Group 20)
Custom
Custom Encryption...
Note:
- Gateways prior to R71 always use the custom encryption settings.
- Suite-B suites and custom GCM encryptions are supported on:
- Gateways of version R71.50
- Gateways from version R75.40
OK
Cancel

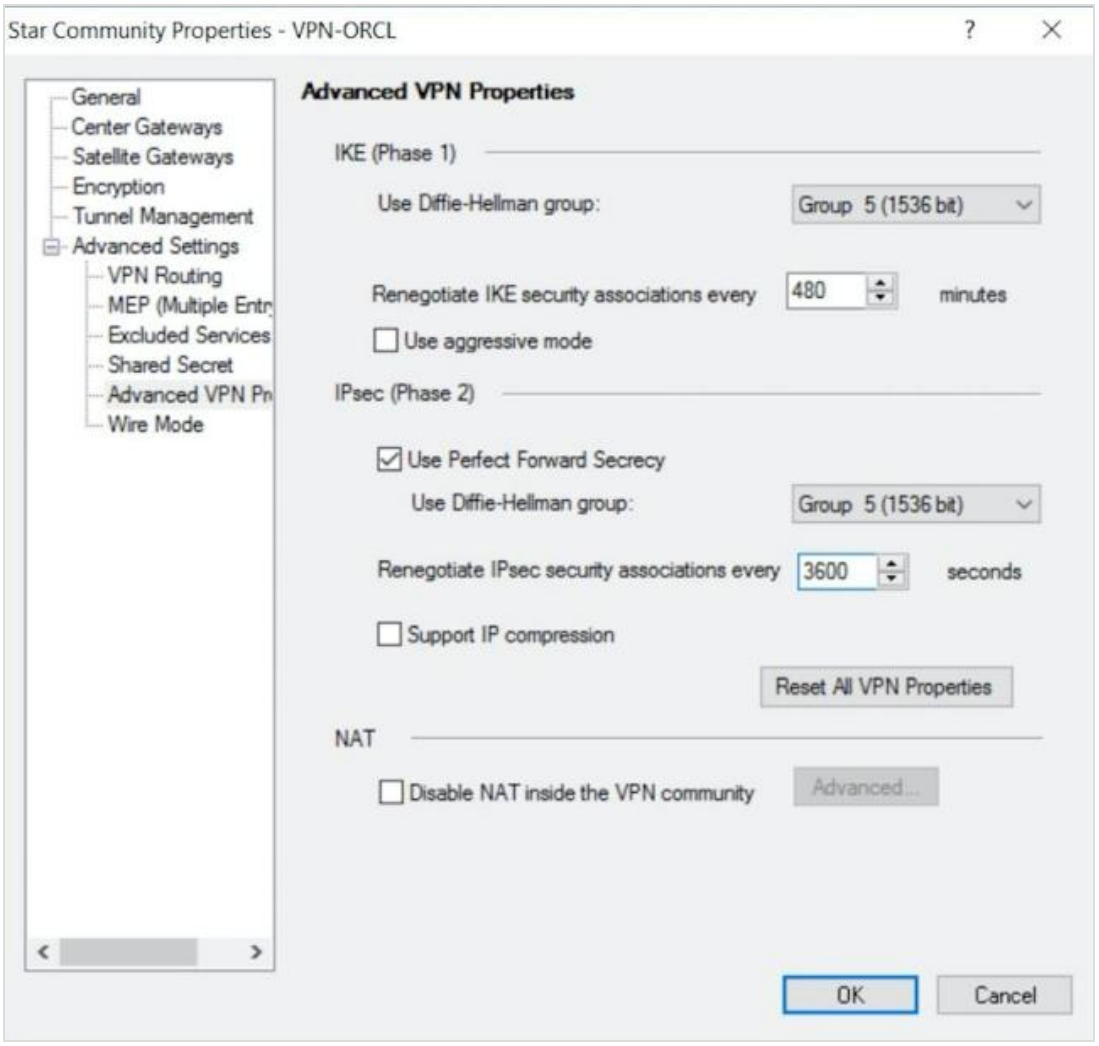


- Under Advanced Settings > Shared Secret, configure pre-shared keys.
    - For each Peer, there is a unique ${sharedSecret#}

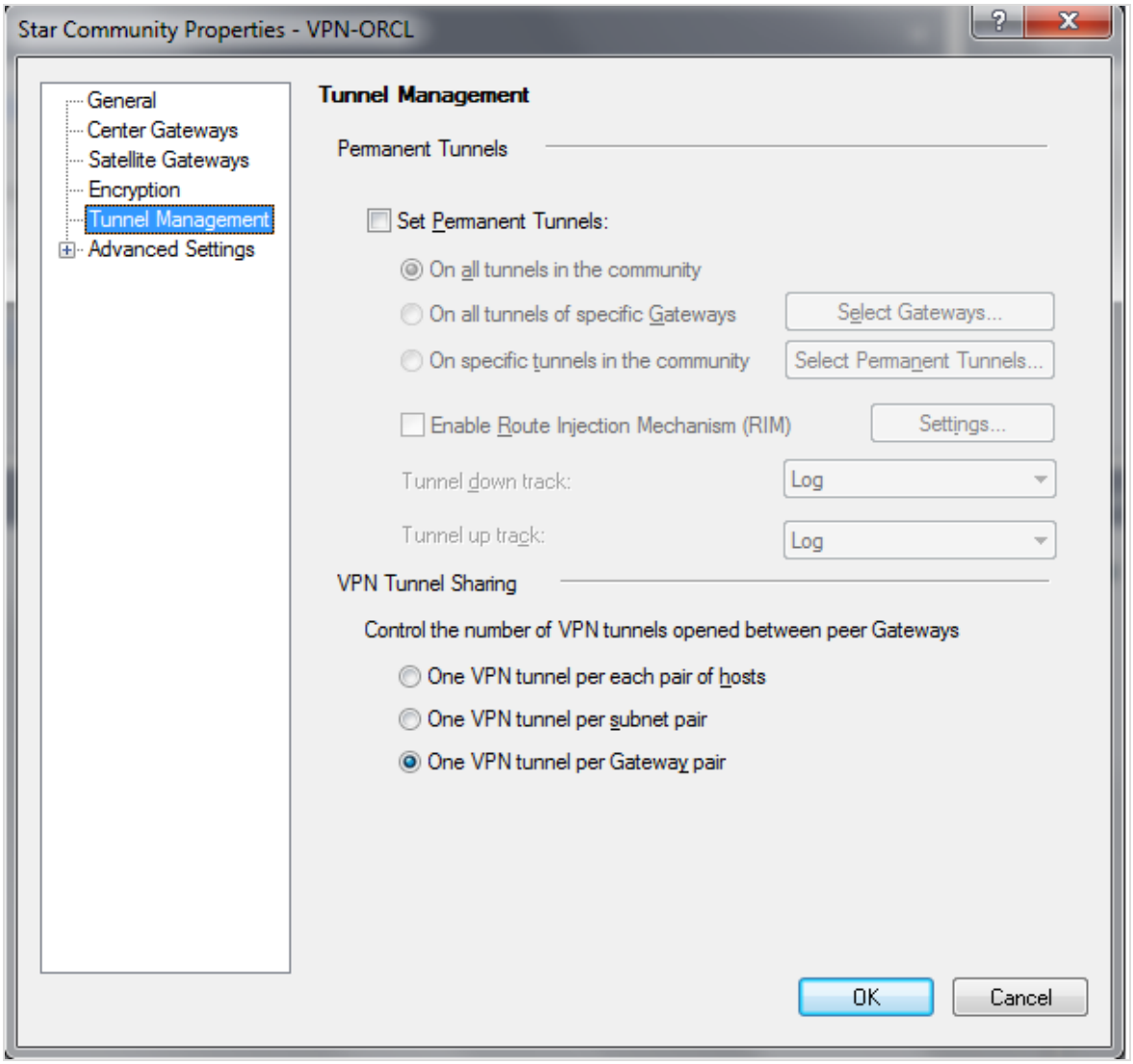


- Under Advanced Settings > Advanced VPN Properties:

- IKE (Phase 1):
  - Use Diffie-Hellman group: Group 5
  - Renegotiate IKE security associations every: 480 minutes
  - Uncheck "Use aggressive mode"
- IPsec (Phase 2)
  - Check "Use Perfect Forward Secrecy"
  - Use Diffie-Hellman group: Group 5
  - Renegotiate IPsec security associations every: 3600 seconds


Star Community Properties - VPN-ORCL
General
Center Gateways
Satellite Gateways
Encryption
Tunnel Management
Advanced Settings
VPN Routing
MEP (Multiple Entry
Excluded Services
Shared Secret
Advanced VPN Pr
Wire Mode
Advanced VPN Properties
IKE (Phase 1)
Use Diffie-Hellman group:
Group 5 (1536 bit)
Renegotiate IKE security associations every
480
minutes
Use aggressive mode
IPsec (Phase 2)
Use Perfect Forward Secrecy
Use Diffie-Hellman group:
Group 5 (1536 bit)
Renegotiate IPsec security associations every
3600
seconds
Support IP compression
Reset All VPN Properties
NAT
Disable NAT inside the VPN community
Advanced...
OK
Cancel

- Under "Tunnel Management", under "VPN Tunnel Sharing", select "One VPN tunnel per Gateway pair".





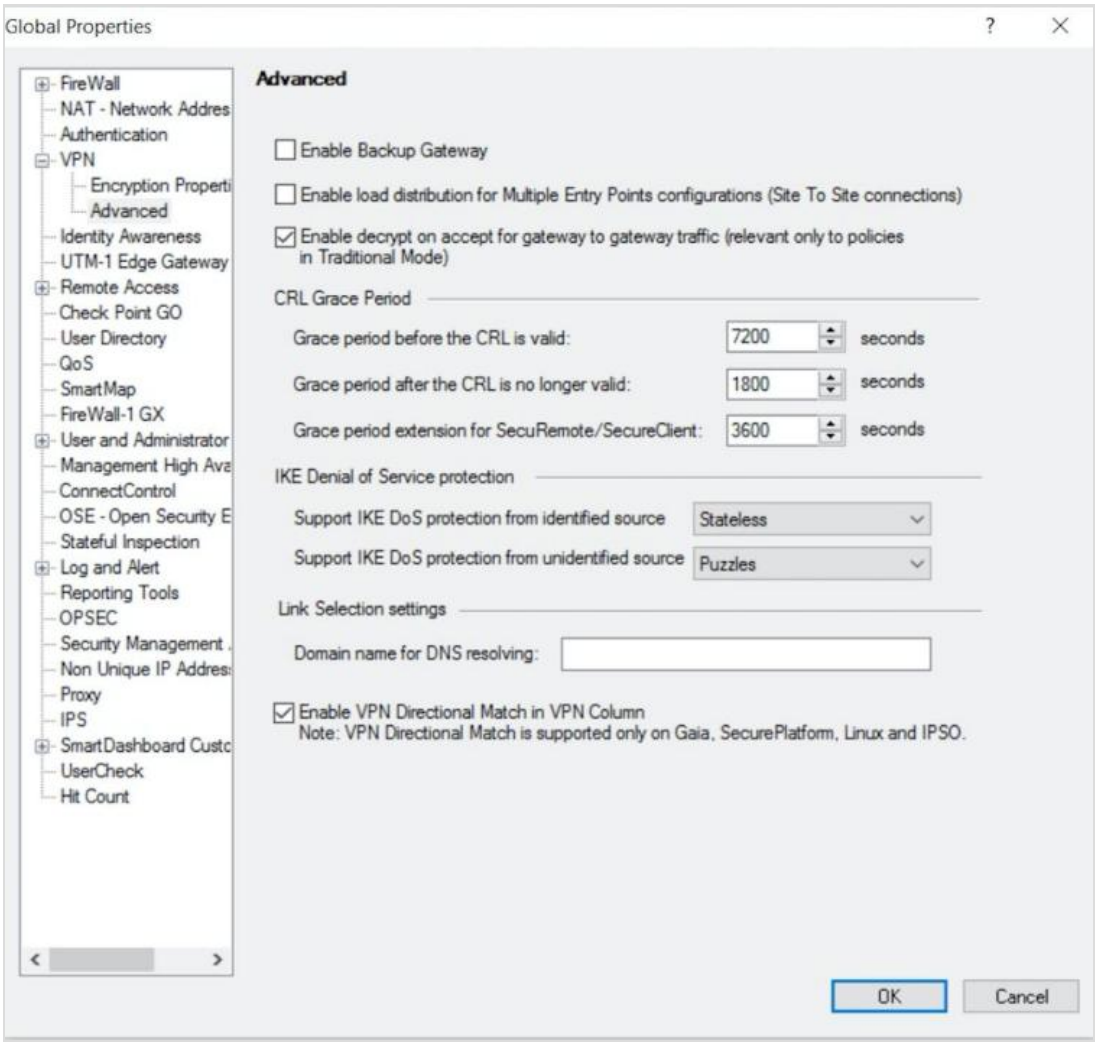
Important

You can select Tunnel Management parameters under both "VPN community" and "Security Gateway Object". **Make sure you select "One VPN tunnel per Gateway pair" under both of these sections, or else you may have connectivity issues due to conflicting settings.**

More details: If there's a conflict between the tunnel properties of a VPN community and a Security Gateway object that is a member of that same community, the stricter setting is followed. For example, a Security Gateway set to "One VPN Tunnel per each pair of hosts" and a community set to "One VPN Tunnel per Gateway pair" would follow "One VPN Tunnel per each pair of hosts".

- Create Firewall Rules: Open Global Properties > VPN > Advanced.

- Check "Enable VPN Directional Match"



- For any firewall rule that matches VPN traffic, add match rules for each direction.



- Install firewall policy



Note

Redundant tunnel configuration is not recommended because the tunnel failover will not work due to absence of an IP address (numbered VTIs or BGP IPSec) that can be used to ping the peer for DPD and failover. If you need redundant tunnels, please contact your Oracle representative.

Cisco ASA Configuration Options

Choose the configuration that suits your situation:

- Route-based configuration: Recommended if your Cisco ASA is running version 9.7.1 or newer.
- Policy-based without VPN filters: Recommended if your Cisco ASA is running a version older than 9.7.1, and you're not already using VPN filters.
- Policy-based with VPN filters: Recommended only if you're already using VPN filters.



Important

Cisco ASA versions 9.7.1 and newer support route-based configuration, which is the recommended method to avoid interoperability issues.

If you want tunnel redundancy with a single Cisco ASA device, you must use route-based configuration. With policy-based configuration, you can configure only a single tunnel between your Cisco ASA and your dynamic routing gateway (DRG), even though Oracle provides configuration information for two tunnels.

Cisco ASA: Route-Based

Cisco ASA versions 9.7.1 and newer support route-based configuration, which is the recommended method to avoid interoperability issues. If you have an earlier version of software, see either Cisco ASA: Policy-Based Without VPN Filters or Cisco ASA: Policy-Based with VPN Filters.



Important

Oracle uses asymmetric routing across the multiple tunnels that make up the IPSec VPN connection. Even if you configure one tunnel as primary and another as backup, traffic from your VCN to your on-premises network can use any tunnel that is "up" on your device. Configure your firewalls accordingly. Otherwise, ping tests or application traffic across the connection will not reliably work.

Parameters from API or Console

Get the following parameters from the Oracle Cloud Infrastructure Console or API.

${ipAddress#}

- Oracle VPN headend IPSec tunnel endpoints. There is one value for each tunnel.
- Example value: 129.146.12.52

${sharedSecret#}

- The IPSec IKE pre-shared-key. There is one value for each tunnel.
- Example value: EXAMPLEDPfAMkD7nTH3SWr6OFabdT6exXn6enSlsKbE

${cpePublicIpAddress}

- The public IP address for the CPE (previously made available to Oracle via the Console). This is the IP address of your outside interface.
- Example Value: 1.2.3.4

Additional Configuration Parameters

The Cisco ASA config requires the following additional variables:

${vcnID}

- A UUID string used to uniquely name some access-lists and object-groups (can also use any other string that does not create a name that conflicts with an existing object-group or access-list).
- Example: oracle-vcn-1

${VcnCidrBlock}

- When creating the VCN, your company selected this CIDR to represent the IP aggregate network for all VCN hosts.
- Example Value: 10.0.0.0/20

${VcnCidrNetwork}** and **${VcnCidrNetmask}

- These are the base address and netmask for the `${VcnCidrBlock}`
- For more information, see: Wikipedia reference for finding CidrNetmask
- Values based on the preceding example `${VcnCidrBlock}`:
  - `${VcnCidrNetwork}`: 10.0.0.0
  - `${VcnCidrNetmask}`: 255.255.0.0

Parameters Discovered from CPE Configuration

The following parameters are based on your current CPE Configuration.

${outsideInterface}

- This is the interface that faces outside of your CPE.
- The outside interface should be able to ping the Oracle VPN headend IPs.

${CpeInsideTunnelIpAddressWithMask#}** and **${OracleInsideTunnelIPAddress#}

These two variables are a pair of IP addresses that you allocate for a given tunnel (one IP address for each end of the tunnel). As shown in the following table, you allocate one pair of addresses per tunnel you configure. The variable for the CPE side of the tunnel includes the mask.

Tunnel	CPE Side of Tunnel	Oracle Side of Tunnel
Tunnel 1	`${CpeInsideTunnelIpAddressWithMask1}` Example: 192.168.1.1 255.255.255.252	`${OracleInsideTunnelIPAddress1}` Example: 192.168.1.2
Tunnel 2	`${CpeInsideTunnelIpAddressWithMask2}` Example: 192.168.1.5 255.255.255.252	`${OracleInsideTunnelIPAddress2}` Example: 192.168.1.6

Config Template Parameter Summary

Each region has multiple Oracle IPSec headends. The following table allows you to set up multiple tunnels on your CPE, each to a corresponding headend. In the table, "User" is you/your company.

Parameter	Source	Example Value
`${ipAddress1}`	Console/API	129.146.12.52
`${sharedSecret1}`	Console/API	(long string)
`${ipAddress2}`	Console/API	129.146.13.52

Parameter	Source	Example Value
`${sharedSecret2}`	Console/API	(long string)
`${cpePublicIpAddress}`	User	1.2.3.4
`${vcnID}`	Console/API/User	1
`${VcnCidrBlock}`	User	10.0.0.0/16
`${VcnCidrNetwork}`	User	10.0.0.0
`${VcnCidrNetmask}`	User	255.255.0.0
`${outsideInterface}`	User CPE	Vlan101
`${outsideInterfaceName}`	User CPE	outside
`${CpeInsideTunnelIpAddressWithMask1}`	User CPE	192.168.1.1 255.255.255.252
`${CpeInsideTunnelIpAddressWithMask2}`	User CPE	192.168.1.5 255.255.255.252
`${OracleInsideTunnelIPAddress1}`	User CPE	192.168.1.2
`${OracleInsideTunnelIPAddress2}`	User CPE	192.168.1.6
`${tunnelInterfaceName1}`	User CPE	tunnel1
`${tunnelInterfaceName2}`	User CPE	tunnel2

ISAKMP Policy Options

Parameter	Recommended Value
ISAKMP protocol version	Version 1
Exchange type	Main mode
Authentication method	Pre-shared keys
Encryption	AES-256-cbc
Authentication algorithm	SHA-1
Diffie-Hellman Group	Group 5
IKE session key lifetime	28800 seconds (8 hours)

IPSec Policy Options

Parameter	Recommended Value
IPSec protocol	ESP, tunnel-mode
Encryption	AES-256-cbc
Authentication algorithm	HMAC-SHA1-96
Diffie-Hellman Group	Group 5
Perfect Forward Secrecy	Enabled
IPSec session key lifetime	3600 seconds (1 hour)

CPE Configuration

IKE and IPSec Policy Configuration

```
crypto ikev1 enable ${outsideInterface}

crypto ikev1 policy 1
  authentication pre-share
  encryption aes-256
  hash sha
  group 5
  lifetime 28800

crypto ipsec ikev1 transform-set oracle-vcn-transform esp-aes-256 esp-sha-hmac

 crypto ipsec profile oracle-vcn-vpn-policy
   set ikev1 transform-set oracle-vcn-transform
   set pfs group5
   set security-association lifetime seconds 3600
```

Tunnel Group Configuration

```
tunnel-group ${ipAddress1} type ipsec-l2l
tunnel-group ${ipAddress1} ipsec-attributes
  ikev1 pre-shared-key ${sharedSecret1}

tunnel-group ${ipAddress2} type ipsec-l2l
tunnel-group ${ipAddress2} ipsec-attributes
 ikev1 pre-shared-key ${sharedSecret2}
```

VTI Interface Configuration

```
interface ${tunnelInterfaceName1}
  nameif ORACLE-VPN1
  ip address ${CpeInsideTunnelIpAddressWithMask1}
  tunnel source interface ${outsideInterfaceName}
  tunnel destination ${ipAddress1}
  tunnel mode ipsec ipv4
  tunnel protection ipsec profile oracle-vcn-vpn-policy

interface ${tunnelInterfaceName2}
  nameif ORACLE-VPN2
  ip address ${CpeInsideTunnelIpAddressWithMask2}
```

```
  tunnel source interface ${outsideInterfaceName}
  tunnel destination ${ipAddress2}
  tunnel mode ipsec ipv4
  tunnel protection ipsec profile oracle-vcn-vpn-policy
```

STATIC ROUTE CONFIGURATION

```
route ORACLE-VPN1 ${VcnCidrNetwork} ${VcnCidrNetmask} ${OracleInsideTunnelIpAddress1} 1 track
route ORACLE-VPN2 ${VcnCidrNetwork} ${VcnCidrNetmask} ${OracleInsideTunnelIpAddress2} 100
```

CONFIGURATION FOR TUNNEL FAILOVER

```
sla monitor 10
  type echo protocol ipIcmpEcho ${ipAddress1}interface outside
  frequency 5
  sla monitor schedule 10 life forever start-time now

track 1 rtr 10 reachability
```

Other Important Device Configuration

- Ensure NAT is disabled for VPN traffic.
- Ensure access-lists on the ASA are configured correctly.
- If you have multiple tunnels up simultaneously, ensure your ASA is configured to handle traffic coming from your VCN/Oracle on *any of the tunnels*, even if one is active and the other is backup. For example, you need to disable ICMP inspection, configure TCP state bypass, and so on. For more details about the appropriate configuration, contact Cisco support.



Warning

Do not use the `originate-only` option with an Oracle IPSec VPN tunnel. It causes the tunnel's traffic to be inconsistently blackholed. The command is only for tunnels between two Cisco devices. Here's an example of the command that you should NOT use for the Oracle IPSec VPN tunnels:

```
crypto map <map name> <sequence number> set connection-type
originate-only
```

Cisco ASA: Policy-Based Without VPN Filters



Important

Cisco ASA versions 9.7.1 and newer support route-based configuration, which is the recommended method to avoid interoperability issues.

If you want tunnel redundancy with a single Cisco ASA device, you must use route-based configuration. With policy-based configuration, you can configure only a single tunnel between your Cisco ASA and your dynamic routing gateway (DRG), even though Oracle provides configuration information for two tunnels.

Parameters from API or Console

Get the following parameters from the Oracle Cloud Infrastructure Console or API.

${ipAddress#}

- Oracle VPN headend IPSec tunnel endpoints. There is one value for each tunnel.
- Example value: 129.146.12.52

${sharedSecret#}

- The IPSec IKE pre-shared-key. There is one value for each tunnel.
- Example value: EXAMPLEDPfAMkD7nTH3SWr6OFabdT6exXn6enSlsKbE

${cpePublicIpAddress}

- The public IP address for the CPE (previously made available to Oracle via the Console). This is the IP address of your outside interface.
- Example Value: 1.2.3.4

Additional Configuration Parameters

The Cisco ASA config requires the following additional variables:

${vcnID}

- A UUID string used to uniquely name some access-lists and object-groups (can also use any other string that does not create a name that conflicts with an existing object-group or access-list).
- Example: oracle-vcn-1

${VcnCidrBlock}

- When creating the VCN, your company selected this CIDR to represent the IP aggregate network for all VCN hosts.
- Example Value: 10.0.0.0/20

${VcnCidrNetwork}** and **${VcnCidrNetmask}

- These are the base address and netmask for the `${VcnCidrBlock}`
- For more information, see: Wikipedia reference for finding CidrNetmask

- Values based on the example `${VcnCidrBlock}` shown above:
  - `${VcnCidrNetwork}`: 10.0.0.0
  - `${VcnCidrNetmask}`: 255.255.0.0

${custCIDR}** and **${custMask}

- To disable NAT between your VCN and your on-premises network, you must define the source IP addresses for packets going through your CPE into the IPSec tunnels.
- Example values:
  - ${custCIDR}: 0.0.0.0
  - ${custMask}: 0.0.0.0

Parameters Discovered from CPE Configuration

The following parameters are based on your current CPE Configuration.

Sample Router Config:

```
interface Vlan100
nameif outside
ip address 192.0.2.1 255.255.255.0
!
interface Vlan101
nameif inside
access-group outbound-acl out interface Vlan101
access-group inbound-acl in interface Vlan100
```

${insideInterface}** and **${outsideInterface}

- These are the interfaces that face the inside and outside of your CPE.
- The outside interface should be able to ping the Oracle VPN headend IPs.
- The inside interface is the one facing your customer premise infrastructure.
- Values based on Sample Router Config above:
  - `${outsideInterface}`: Vlan100
  - `${insideInterface}`: Vlan101

${insideInterfaceName}** and **${outsideInterfaceName}

- These are the "nameif" values for the inside and outside interfaces.
- Values based on Sample Router Config above:
  - `${insideInterfaceName}`: inside
  - `${outsideInterfaceName}`: outside

${inboundAclName}** and **${outboundAclName}

- You likely also have access-lists configured to control traffic in and out of your inside and outside interfaces.
- Values based on Sample Router Config above:
  - `${inboundAclName}`: inbound-acl
  - `${outboundAclName}`: outbound-acl

Config Template Parameter Summary

Each region has multiple Oracle IPSec headends. The template below will allow setting up multiple tunnels on your CPE, each to a corresponding headend. In the table below, "User" is you/your company.

Parameter	Source	Example Value
`${ipAddress1}`	Console/API	129.146.12.52
`${sharedSecret1}`	Console/API	(long string)
`${ipAddress2}`	Console/API	129.146.13.52
`${sharedSecret2}`	Console/API	(long string)
`${cpePublicIpAddress}`	User	1.2.3.4
`${vcnID}`	Console/API/User	1
`${VcnCidrBlock}`	User	10.0.0.0/16

Parameter	Source	Example Value
`${VcnCidrNetwork}`	User	10.0.0.0
`${VcnCidrNetmask}`	User	255.255.0.0
`${VcnHostIp}`	User	IP address of a host inside your VCN. Preferably this host can reply to ICMP Echo requests. Example: 10.0.2.7
`${custCIDR}`	User	0.0.0.0
`${custMask}`	User	0.0.0.0
`${insideInterface}`	User CPE	Vlan101
`${outsideInterface}`	User CPE	Vlan100
`${insideInterfaceName}`	User CPE	inside
`${outsideInterfaceName}`	User CPE	outside
`${inboundAclName}`	User CPE	inbound-acl
`${outboundAclName}`	User CPE	outbound-acl
`${cryptoMapAclName}`	User CPE	orcl-acl

ISAKMP Policy Options

Parameter	Recommended Value
ISAKMP protocol version	Version 1
Exchange type	Main mode
Authentication method	Pre-shared keys

Parameter	Recommended Value
Encryption	AES-256-cbc
Authentication algorithm	SHA-1
Diffie-Hellman Group	Group 5
IKE session key lifetime	28800 seconds (8 hours)

IPSec Policy Options

Parameter	Recommended Value
IPSec protocol	ESP, tunnel-mode
Encryption	AES-256-cbc
Authentication algorithm	HMAC-SHA1-96
Diffie-Hellman Group	Group 5
Perfect Forward Secrecy	Enabled
IPSec session key lifetime	3600 seconds (60 minutes)

CPE Configuration

NETWORK OBJECT DEFINITION

The ASA uses configuration objects to identify IP networks that are used in multiple locations.

DEFINE OBJECT THAT REPRESENTS YOUR ORACLE VCN

This object group is used by the IPSec policies to understand what IP addresses belong to your Oracle VCN, so that they can be encrypted and sent inside the correct IPSec tunnel.

```
object network oracle-vcn-${vcnID}
subnet ${VcnCidrNetwork} ${VcnCidrNetmask}
```

DEFINE OBJECT THAT REPRESENTS YOUR CUSTOMER PREMISE NETWORK

This object may already be present on your ASA. A common name would match the interface name of your "inside" interface. You might have multiple "subnet" entries in this object-group. One for each aggregate subnet you want to allow this IPSec tunnel to be used for traffic to and from your Oracle VCN.

```
object network ${insideInterfaceName}
subnet ${custCIDR} ${custMask}
```

DISABLE NAT FOR VPN TRAFFIC

If you are using NAT for traffic between your inside and outside interfaces, you might need to disable NAT for traffic between your on-premises network and the Oracle VCN.

```
nat (${insideInterfaceName},${outsideInterfaceName}) source static ${insideInterfaceName}
${insideInterfaceName} destination static oracle-vcn-${vcnID} oracle-vcn-${vcnID}
```

PERMIT TRAFFIC BETWEEN YOUR CPE AND YOUR ORACLE VCN

Assuming there is an access-list controlling traffic in and out of your Internet facing interface, you will need to permit traffic between your CPE and the Oracle VPN headend.

WARNING: The new ACL entry you add to permit the traffic between your CPE and VPN headend needs to be above any deny statements you might have in your existing access-list:

```
access-list ${outboundAclName} extended permit ip host ${ipAddress1} host ${cpePublicIpAddress}
access-list ${outboundAclName} extended permit ip host ${ipAddress2} host ${cpePublicIpAddress}
```

NOTE: You can be more restrictive in your ACL if you wish by only permitting the following:

- udp port 500 : isakmp
- esp : ipsec encapsulated secure payload

CRYPTO MAP ACL CONFIGURATION

The following access list "orcl-acl" will be associated with the IPSec security association using the "crypto-map" command.

```
access-list ${cryptoMapAclName} extended permit ip any ${VcnCidrNetwork} ${VcnCidrNetmask}
```



Important

You must initiate the tunnel. If the Oracle IPSec gateway initiates the tunnel, it only supports sending the crypto map equivalent of "ip any any". As the responder, it accepts only a single entry in the crypto map: `ip any ${VcnCidrNetwork} ${VcnCidrNetmask}` as shown in the preceding command.

ISAKMP Policy Configuration

See the ASA Command Reference.

```
crypto ikev1 enable ${outsideInterfaceName}
crypto ikev1 policy 1
authentication pre-share
encryption aes-256
hash sha
group 5
lifetime 28800
```

Base VPN Policy Configuration

This configuration sets the base values for the IPSec tunnels.

```
group-policy oracle-vcn-vpn-policy internal
group-policy oracle-vcn-vpn-policy attributes
vpn-idle-timeout none
vpn-session-timeout none
vpn-tunnel-protocol ikev1
```

See the relevant Cisco documentation:

- group-policy

IPSec Configuration

See the relevant Cisco reference documentation.



Warning

Make sure your crypto map sequence numbers do not overlap with existing crypto maps.

Do not use the `originate-only` option with an Oracle IPSec VPN tunnel. It causes the tunnel's traffic to be inconsistently blackholed. The command is only for tunnels between two Cisco devices. Here's an example of the command that you should NOT use for the Oracle IPSec VPN tunnels:

```
crypto map <map name> <sequence number> set connection-type
originate-only
```

```
crypto ipsec ikev1 transform-set oracle-vcn-transform esp-aes-256 esp-sha-hmac
crypto map oracle-${ipAddress1}-map-v1 1 match address ${cryptoMapAclName}
crypto map oracle-${ipAddress1}-map-v1 1 set pfs group5
crypto map oracle-${ipAddress1}-map-v1 1 set peer ${ipAddress1}
crypto map oracle-${ipAddress1}-map-v1 1 set ikev1 transform-set oracle-vcn-transform
crypto map oracle-${ipAddress1}-map-v1 1 set security-association lifetime seconds 3600
crypto map oracle-${ipAddress1}-map-v1 interface outside

crypto map oracle-${ipAddress2}-map-v1 2 match address ${cryptoMapAclName}
crypto map oracle-${ipAddress2}-map-v1 2 set pfs group5
crypto map oracle-${ipAddress2}-map-v1 2 set peer ${ipAddress2}
crypto map oracle-${ipAddress2}-map-v1 2 set ikev1 transform-set oracle-vcn-transform
crypto map oracle-${ipAddress2}-map-v1 2 set security-association lifetime seconds 3600
crypto map oracle-${ipAddress2}-map-v1 interface outside
```

PER TUNNEL IPSEC CONFIGURATION

This configuration matches the group policy with an Oracle VPN headend endpoint.

```
tunnel-group ${ipAddress1} type ipsec-l2l
tunnel-group ${ipAddress1} general-attributes
default-group-policy oracle-vcn-vpn-policy
tunnel-group ${ipAddress1} ipsec-attributes
ikev1 pre-shared-key ${sharedSecret1}
```

```
tunnel-group ${ipAddress2} type ipsec-l2l
tunnel-group ${ipAddress2} general-attributes
default-group-policy oracle-vcn-vpn-policy
tunnel-group ${ipAddress2} ipsec-attributes
ikev1 pre-shared-key ${sharedSecret2}
```

SLA Monitoring Configuration

The Cisco ASA device doesn't establish a tunnel if there's no interesting traffic trying to pass through the tunnel. You must configure SLA monitoring on your device so that the tunnel remains up at all times. You must allow ICMP on the outside interface. Make sure that the SLA monitor number is unique.

```
sla monitor 1
type echo protocol ipIcmpEcho ${VcnHostIp} interface outside
frequency 5
sla monitor schedule 1 life forever start-time now
icmp permit any ${outsideInterface}
```

Optional Config Where VPN Traffic Might Enter One Tunnel and Exit Another

If the VPN traffic enters an interface with the same security level as an interface towards the packets next-hop, you will need to allow that traffic. By default, the packets between interfaces with identical security levels on your ASA will be dropped.

See the relevant Cisco reference documentation.

```
same-security-traffic permit inter-interface
```

Dealing with Tunnel MTU and Path MTU Discovery

Option 1 - TCP MSS Adjust

Because the maximum transmission unit (packet size) through the IPSec tunnel is less than 1500 bytes, we need to either fragment packets that are too big to fit through the tunnel or we need to signal back to the hosts communicating through the tunnel that they need to send smaller packets.

You can configure the Cisco ASA to change the maximum segment size (MSS) for any new TCP flows through the tunnel. The ASA will look at any TCP packets where the SYN flag is set and

change the MSS value to the configured value. This might help new TCP flows avoid having to use path maximum transmission unit discovery (pmtud).

```
sysopt connection tcpmss 1387
```

References:

- RFC 1191
- Relevant Cisco reference documentation

OPTION 2 - CLEAR/SET DON'T FRAGMENT BIT

Path MTU Discovery requires that all TCP packets have the Don't Fragment (DF) bit set. When the packet arrives on the ASA, if it is too big to go through the tunnel and the DF bit is set, the ASA will drop the packet and send an ICMP packet back to the sender indicating that the packet was too big to fit through the tunnel. There are three options on the ASA for how to handle the DF bit (pick one of the options):

- Set the DF bit: If the DF bit is not already set and the packet is too big, will set the DF bit, causing the ASA to drop the the packet and send an ICMP error message back to the sender. (Recommended)

  ```
  crypto ipsec df-bit set-df ${outsideInterfaceName}
  ```

- Clear the DF bit: Will allow the ASA to fragment the packet and send it to the end host in Oracle Cloud Infrastructure to reassemble the packet.

  ```
  crypto ipsec df-bit clear-df ${outsideInterfaceName}
  ```

- Ignore (copy) the DF bit: If the packet is too big, and the DF bit is set, will drop the packet and send error message to sender, if the DF bit is not set, will fragment the packet and send to Oracle Cloud Infrastructure.

  ```
  crypto ipsec df-bit copy-df ${outsideInterfaceName}
  ```

References:

- Path MTU Discovery on Wikipedia
- Relevant Cisco reference documentation

Cisco ASA: Policy-Based with VPN Filters



Important

Cisco ASA versions 9.7.1 and newer support route-based configuration, which is the recommended method to avoid interoperability issues.

If you want tunnel redundancy with a single Cisco ASA device, you must use route-based configuration. With policy-based configuration, you can configure only a single tunnel between your Cisco ASA and your dynamic routing gateway (DRG), even though Oracle provides configuration information for two tunnels.

Parameters from API or Console

Get the following parameters from the Oracle Cloud Infrastructure Console or API.

${ipAddress#}

- Oracle VPN headend IPSec tunnel endpoints. There is one value for each tunnel.
- Example value: 129.146.12.52

${sharedSecret#}

- The IPSec IKE pre-shared-key. There is one value for each tunnel.
- Example value: EXAMPLEDPfAMkD7nTH3SWr6OFabdT6exXn6enSlsKbE

${cpePublicIpAddress}

- The public IP address for the CPE (previously made available to Oracle via the Console). This is the IP address of your outside interface.
- Example Value: 1.2.3.4

Additional Configuration Parameters

The Cisco ASA config requires the following additional variables:

${vcnID}

- A UUID string used to uniquely name some access-lists and object-groups (can also use any other string that does not create a name that conflicts with an existing object-group or access-list).
- Example: oracle-vcn-1

${VcnCidrBlock}

- When creating the VCN, your company selected this CIDR to represent the IP aggregate network for all VCN hosts.
- Example Value: 10.0.0.0/20

${VcnCidrNetwork}** and **${VcnCidrNetmask}

- These are the base address and netmask for the `${VcnCidrBlock}`
- For more information, see: Wikipedia reference for finding CidrNetmask
- Values based on the example `${VcnCidrBlock}` shown above:
  - `${VcnCidrNetwork}`: 10.0.0.0
  - `${VcnCidrNetmask}`: 255.255.0.0

${custCIDR}** and **${custMask}

- To disable NAT between your VCN and your on-premises network, you must define the source IP addresses for packets going through your CPE into the IPSec tunnels.
- Example values:
  - ${custCIDR}: 0.0.0.0
  - ${custMask}: 0.0.0.0

Parameters Discovered from CPE Configuration

The following parameters are based on your current CPE Configuration.

Sample Router Config:

```
interface Vlan100
 nameif outside
 ip address 192.0.2.1 255.255.255.0
!
interface Vlan101
 nameif inside
access-group outbound-acl out interface Vlan101
access-group inbound-acl in interface Vlan100
```

${insideInterface}** and **${outsideInterface}

- These are the interfaces that face the inside and outside of your CPE.
- The outside interface should be able to ping the Oracle VPN headend IPs.
- The inside interface is the one facing your customer premise infrastructure.
- Values based on Sample Router Config above:
  - `${outsideInterface}`: Vlan100
  - `${insideInterface}`: Vlan101

${insideInterfaceName}** and **${outsideInterfaceName}

- These are the "nameif" values for the inside and outside interfaces.
- Values based on Sample Router Config above:
  - `${insideInterfaceName}`: inside
  - `${outsideInterfaceName}`: outside

${inboundAclName}** and **${outboundAclName}

- You likely also have access-lists configured to control traffic in and out of your inside and outside interfaces.
- Values based on Sample Router Config above:
  - `${inboundAclName}`: inbound-acl
  - `${outboundAclName}`: outbound-acl

Config Template Parameter Summary

Each region has multiple Oracle IPSec headends. The template below will allow setting up multiple tunnels on your CPE, each to a corresponding headend. In the table below, "User" is you/your company.

Parameter	Source	Example Value
`${ipAddress1}`	Console/API	129.146.12.52
`${sharedSecret1}`	Console/API	(long string)
`${ipAddress2}`	Console/API	129.146.13.52
`${sharedSecret2}`	Console/API	(long string)
`${cpePublicIpAddress}`	User	1.2.3.4
`${vcnID}`	Console/API/User	1
`${VcnCidrBlock}`	User	10.0.0.0/16
`${VcnCidrNetwork}`	User	10.0.0.0
`${VcnCidrNetmask}`	User	255.255.0.0
`${VcnHostIp}`	User	IP address of a host inside your VCN. Preferably this host can reply to ICMP Echo requests. Example: 10.0.2.7
`${custCIDR}`	User	0.0.0.0
`${custMask}`	User	0.0.0.0
`${insideInterface}`	User CPE	Vlan101
`${outsideInterface}`	User CPE	Vlan100
`${insideInterfaceName}`	User CPE	inside

Parameter	Source	Example Value
`${outsideInterfaceName}`	User CPE	outside
`${inboundAclName}`	User CPE	inbound-acl
`${outboundAclName}`	User CPE	outbound-acl
`${cryptoMapAclName}`	User CPE	orcl-acl
`${vpnFilterAclName}`	User CPE	orcl-filter

ISAKMP Policy Options

Parameter	Recommended Value
ISAKMP protocol version	Version 1
Exchange type	Main mode
Authentication method	Pre-shared keys
Encryption	AES-256-cbc
Authentication algorithm	SHA-1
Diffie-Hellman Group	Group 5
IKE session key lifetime	28800 seconds (8 hours)

IPSec Policy Options

Parameter	Recommended Value
IPSec protocol	ESP, tunnel-mode
Encryption	AES-256-cbc

Parameter	Recommended Value
Authentication algorithm	HMAC-SHA1-96
Diffie-Hellman Group	Group 5
Perfect Forward Secrecy	Enabled
IPSec session key lifetime	3600 seconds (60 minutes)

CPE Configuration

Network Object Definition

The ASA uses configuration objects to identify IP networks that are used in multiple locations.

Define Object that Represents Your Oracle VCN

This object group is used by the IPSec policies to understand what IP addresses belong to your Oracle VCN, so that they can be encrypted and sent inside the correct IPSec tunnel.

```
object network oracle-vcn-${vcnID}
 subnet ${VcnCidrNetwork} ${VcnCidrNetmask}
```

Define Object that Represents Your Customer Premise Network

This object may already be present on your ASA. A common name would match the interface name of your "inside" interface. You might have multiple "subnet" entries in this object-group. One for each aggregagte subnet you want to allow this IPSec tunnel to be used for traffic to and from your Oracle VCN.

```
object network ${insideInterfaceName}
 subnet ${custCIDR} ${custMask}
```

Disable NAT for VPN traffic

If you are using NAT for traffic between your inside and outside interfaces, you might need to disable NAT for traffic between your on-premises network and the Oracle VCN.

```
nat (${insideInterfaceName},${outsideInterfaceName}) source static ${insideInterfaceName}
${insideInterfaceName} destination static oracle-vcn-${vcnID} oracle-vcn-${vcnID}
```

PERMIT TRAFFIC BETWEEN YOUR CPE AND YOUR ORACLE VCN

Assuming there is an access-list controlling traffic in and out of your Internet facing interface, you will need to permit traffic between your CPE and the Oracle VPN headend.

WARNING: The new ACL entry you add to permit the traffic between your CPE and VPN headend needs to be above any deny statements you might have in your existing access-list:

```
access-list ${outboundAclName} extended permit ip host ${ipAddress1} host ${cpePublicIpAddress}
access-list ${outboundAclName} extended permit ip host ${ipAddress2} host ${cpePublicIpAddress}
```

NOTE: You can be more restrictive in your ACL if you wish by only permitting the following:

- udp port 500 : isakmp
- esp : ipsec encapsulated secure payload

VPN FILTER ACL CONFIGURATION

The following access list "orcl-acl" will be associated with the IPSec security association using the "crypto-map" command.

```
access-list ${cryptoMapAclName} extended permit ip any any
```

The following ACL will be used in the VPN filter to restrict the actual traffic that will be permitted through the tunnels. See Base VPN Policy Configuration for details.

```
access-list ${vpnFilterAclName} extended permit ip ${VcnCidrNetwork} ${VcnCidrNetmask} ${custCIDR}
${custMask}
```

According to Cisco: if you have an ACL that is used for a vpn-filter, do NOT also use it for an interface access-group.

See the relevant Cisco reference documentation.

ISAKMP POLICY CONFIGURATION

See the ASA Command Reference.

```
crypto ikev1 enable ${outsideInterfaceName}
crypto ikev1 policy 1
 authentication pre-share
 encryption aes-256
 hash sha
```

```
group 5
lifetime 28800
```

Base VPN Policy Configuration

This configuration sets the base values for the IPSec tunnels.

It also restricts what traffic will be allowed over the tunnels via the "vpn filter" command. By default all traffic is denied by the filter.

```
group-policy oracle-vcn-vpn-policy internal
group-policy oracle-vcn-vpn-policy attributes
 vpn-idle-timeout none
 vpn-session-timeout none
 vpn-tunnel-protocol ikev1
 vpn-filter value ${vpnFilterAclName}
```

See the relevant Cisco documentation:

- group-policy
- VPN filter

IPSec Configuration

See the relevant Cisco reference documentation.



Warning

Make sure your crypto map sequence numbers do not overlap with existing crypto maps.

Do not use the `originate-only` option with an Oracle IPSec VPN tunnel. It causes the tunnel's traffic to be inconsistently blackholed. The command is only for tunnels between two Cisco devices. Here's an example of the command that you should NOT use for the Oracle IPSec VPN tunnels:

```
crypto map <map name> <sequence number> set connection-type
originate-only
```

```
crypto ipsec ikev1 transform-set oracle-vcn-transform esp-aes-256 esp-sha-hmac
crypto map oracle-${ipAddress1}-map-v1 1 match address ${cryptoMapAclName}
crypto map oracle-${ipAddress1}-map-v1 1 set pfs group5
crypto map oracle-${ipAddress1}-map-v1 1 set peer ${ipAddress1}
crypto map oracle-${ipAddress1}-map-v1 1 set ikev1 transform-set oracle-vcn-transform
crypto map oracle-${ipAddress1}-map-v1 1 set security-association lifetime seconds 3600
crypto map oracle-${ipAddress1}-map-v1 interface outside

crypto map oracle-${ipAddress2}-map-v1 2 match address ${cryptoMapAclName}
crypto map oracle-${ipAddress2}-map-v1 2 set pfs group5
crypto map oracle-${ipAddress2}-map-v1 2 set peer ${ipAddress2}
crypto map oracle-${ipAddress2}-map-v1 2 set ikev1 transform-set oracle-vcn-transform
crypto map oracle-${ipAddress2}-map-v1 2 set security-association lifetime seconds 3600
crypto map oracle-${ipAddress2}-map-v1 interface outside
```

PER TUNNEL IPSEC CONFIGURATION

This configuration matches the group policy with an Oracle VPN headend endpoint.

```
tunnel-group ${ipAddress1} type ipsec-l2l
tunnel-group ${ipAddress1} general-attributes
 default-group-policy oracle-vcn-vpn-policy
tunnel-group ${ipAddress1} ipsec-attributes
 ikev1 pre-shared-key ${sharedSecret1}
```

```
tunnel-group ${ipAddress2} type ipsec-l2l
tunnel-group ${ipAddress2} general-attributes
 default-group-policy oracle-vcn-vpn-policy
tunnel-group ${ipAddress2} ipsec-attributes
 ikev1 pre-shared-key ${sharedSecret2}
```

SLA Monitoring Configuration

The Cisco ASA device doesn't establish a tunnel if there's no interesting traffic trying to pass through the tunnel. You must configure SLA monitoring on your device so that the tunnel remains up at all times. You must allow ICMP on the outside interface. Make sure that the SLA monitor number is unique.

```
sla monitor 1
 type echo protocol ipIcmpEcho ${VcnHostIp} interface outside
 frequency 5
sla monitor schedule 1 life forever start-time now
icmp permit any ${outsideInterface}
```

Optional Config Where VPN Traffic Might Enter One Tunnel and Exit Another

If the VPN traffic enters an interface with the same security level as an interface towards the packets next-hop, you will need to allow that traffic. By default, the packets between interfaces with identical security levels on your ASA will be dropped.

See the relevant Cisco reference documentation.

```
same-security-traffic permit inter-interface
```

Dealing with Tunnel MTU and Path MTU Discovery

Option 1 - TCP MSS Adjust

Because the maximum transmission unit (packet size) through the IPSec tunnel is less than 1500 bytes, we need to either fragment packets that are too big to fit through the tunnel or we need to signal back to the hosts communicating through the tunnel that they need to send smaller packets.

You can configure the Cisco ASA to change the maximum segment size (MSS) for any new TCP flows through the tunnel. The ASA will look at any TCP packets where the SYN flag is set and

change the MSS value to the configured value. This might help new TCP flows avoid having to use path maximum transmission unit discovery (pmtud).

```
sysopt connection tcpmss 1387
```

References:

- RFC 1191
- Relevant Cisco reference documentation

OPTION 2 - CLEAR/SET DON'T FRAGMENT BIT

Path MTU Discovery requires that all TCP packets have the Don't Fragment (DF) bit set. When the packet arrives on the ASA, if it is too big to go through the tunnel and the DF bit is set, the ASA will drop the packet and send an ICMP packet back to the sender indicating that the packet was too big to fit through the tunnel. There are three options on the ASA for how to handle the DF bit (pick one of the options):

- Set the DF bit: If the DF bit is not already set and the packet is too big, will set the DF bit, causing the ASA to drop the the packet and send an ICMP error message back to the sender. (Recommended)

  ```
  crypto ipsec df-bit set-df ${outsideInterfaceName}
  ```

- Clear the DF bit: Will allow the ASA to fragment the packet and send it to the end host in Oracle Cloud Infrastructure to reassemble the packet.

  ```
  crypto ipsec df-bit clear-df ${outsideInterfaceName}
  ```

- Ignore (copy) the DF bit: If the packet is too big, and the DF bit is set, will drop the packet and send error message to sender, if the DF bit is not set, will fragment the packet and send to Oracle Cloud Infrastructure.

  ```
  crypto ipsec df-bit copy-df ${outsideInterfaceName}
  ```

References:

- Path MTU Discovery on Wikipedia
- Relevant Cisco reference documentation

Cisco IOS

This configuration was validated using a Cisco 2921 running Cisco IOS Version 15.4(3)M3.



Important

Oracle uses asymmetric routing across the multiple tunnels that make up the IPSec VPN connection. Even if you configure one tunnel as primary and another as backup, traffic from your VCN to your on-premises network can use any tunnel that is "up" on your device. Configure your firewalls accordingly. Otherwise, ping tests or application traffic across the connection will not reliably work.

Parameters from API or Console

Get the following parameters from the Oracle Cloud Infrastructure Console or API.

${ipAddress#}

- Oracle VPN headend IPSec tunnel endpoints. There is one value for each tunnel.
- Example value: 129.146.12.52

${sharedSecret#}

- The IPSec IKE pre-shared-key. There is one value for each tunnel.
- Example value: EXAMPLEDPfAMkD7nTH3SWr6OFabdT6exXn6enSlsKbE

${cpePublicIpAddress}

- The public IP address for the CPE (previously made available to Oracle via the Console).

${VcnCidrBlock}

- When creating the VCN, your company selected this CIDR to represent the IP aggregate network for all VCN hosts.
- Example Value: 10.0.0.0/20

Parameters Based on Current CPE Configuration and State

The following parameters need to be discovered by examining your CPE's current configuration and finding valid parameter values that do not overlap with the current CPE configuration.

${tunnelNumber#} - one per tunnel

- The Cisco IOS uses the **Tunnel** interfaces for "Virtual Tunnel Interfaces" (vti) IPSec tunnels.
- Command to find value: `show run | inc interface Tunnel`
- You will need one unused unit number per tunnel.
- Example Values: 1, 2

Config Template Parameter Summary

Each region has multiple Oracle IPSec headends. The template below will allow setting up multiple tunnels on your CPE, each to a corresponding headend. In the table below, "User" is you/your company.

Parameter	Source	Example Value
`${ipAddress1}`	Console/API	129.146.12.52
`${sharedSecret1}`	Console/API	(long string)
`${ipAddress2}`	Console/API	129.146.13.52
`${sharedSecret2}`	Console/API	(long string)
`${cpePublicIpAddress}`	User	1.2.3.4

Parameter	Source	Example Value
`${tunnelNumber1}`	User	1
`${tunnelNumber2}`	User	2
`${VcnCidrBlock}`	User	10.0.0.0/16

ISAKMP Policy Options

Parameter	Recommended Value
ISAKMP protocol version	Version 1
Exchange type	Main mode
Authentication method	Pre-shared keys
Encryption	AES-256-cbc
Authentication algorithm	SHA-384
Diffie-Hellman Group	Group 5
IKE session key lifetime	28800 seconds (8 hours)

IPSec Policy Options

Parameter	Recommended Value
IPSec protocol	ESP, tunnel-mode
Encryption	AES-256-cbc
Authentication algorithm	HMAC-SHA1-96
Diffie-Hellman Group	Group 5

Parameter	Recommended Value
Perfect Forward Secrecy	Enabled
IPSec session key lifetime	3600 seconds (1 hour)

Security Parameter Index

The values for the Security Parameter Index (SPI) depend on whether your CPE supports route-based tunnels or policy-based tunnels. For more information about the correct SPI values to use, see Route-Based Versus Policy-Based IPSec.

CPE Configuration

Pre-shared-key Configuration

Add the pre-shared-keys to your configuration:

```
crypto keyring oracle-vpn-${ipAddress1}
  local-address ${cpePublicIpAddress}
  pre-shared-key address ${ipAddress1} key ${sharedSecret1}
crypto keyring oracle-vpn-${ipAddress2}
  local-address ${cpePublicIpAddress}
  pre-shared-key address ${ipAddress2} key ${sharedSecret2}
```

Configure Basic ISAKMP Options

These are optional configuration options:

```
crypto logging session
crypto isakmp fragmentation
crypto isakmp keepalive 10 10
crypto ipsec df-bit clear
crypto ipsec security-association replay window-size 128
```

Configure Base ISAKMP and IPSec Policies

```
crypto isakmp policy 1
 encr aes 256
 hash sha384
 authentication pre-share
```

```
 group 5
 lifetime 28800
crypto ipsec transform-set oracle_vpn_transform esp-aes 256 esp-sha-hmac
 mode tunnel
crypto ipsec profile oracle_vpn
 set security-association dfbit clear
 set transform-set oracle_vpn_transform
 set pfs group5
```

CONFIGURE ISAKMP AND IPSEC PEERS

```
crypto isakmp profile oracle_vpn_${ipAddress1}
   keyring oracle-vpn-${ipAddress1}
   self-identity address
   match identity address ${ipAddress1} 255.255.255.255
   keepalive 10 retry 10
crypto isakmp profile oracle_vpn_${ipAddress2}
   keyring oracle-vpn-${ipAddress2}
   self-identity address
   match identity address ${ipAddress2} 255.255.255.255
   keepalive 10 retry 10
```

CONFIGURE VIRTUAL TUNNEL INTERFACES

```
interface Tunnel${tunnelNumber1}
 ip tcp adjust-mss 1387
 tunnel source ${cpePublicIpAddress}
 tunnel mode ipsec ipv4
 tunnel destination ${ipAddress1}
 tunnel protection ipsec profile oracle_vpn
!
interface Tunnel${tunnelNumber2}
 tunnel source ${cpePublicIpAddress}
 tunnel mode ipsec ipv4
 tunnel destination ${ipAddress2}
 tunnel protection ipsec profile oracle_vpn
```

UPDATE ANY INTERNET FACING ACCESS LIST TO ALLOW IPSEC AND ISAKMP PACKETS

In order for the tunnels to come up, you need to open up udp port 500 and protocol esp to the interface with your CPE Public IP Address. Example:

```
interface GigabitEthernet0/1
 description INTERNET
 ip address ${cpePublicIpAddress} 255.255.255.252
ip access-group INTERNET-INGRESS in
 duplex auto
 speed auto
!

ip access-list extended INTERNET-INGRESS
 permit udp host ${ipAddress1} host ${cpePublicIpAddress} eq isakmp
 permit esp host ${ipAddress1} host ${cpePublicIpAddress}
 permit udp host ${ipAddress2} host ${cpePublicIpAddress} eq isakmp
 permit esp host ${ipAddress2} host ${cpePublicIpAddress}
 permit icmp any any echo
 permit icmp any any echo-reply
 permit icmp any any unreachable
```

CPE TO VCN STATIC ROUTES

The IPSec tunnels require static routes to get traffic to go through them. The routes are configured to point down the tunnel interfaces. If the IPSec tunnel is down, the Cisco router will stop using that route. You should redistribute these routes into your customer premise network.

```
ip route ${vcnCidrBlock} 255.0.0.0 Tunnel${tunnelNumber1}
ip route ${vcnCidrBlock} 255.0.0.0 Tunnel${tunnelNumber2}
```

FortiGate

This configuration was validated using a FortiGate-VM running v5.4.1,build1064 (GA).



Important

Oracle uses asymmetric routing across the multiple tunnels that make up the IPSec VPN connection. Even if you configure one tunnel as primary and another as backup, traffic from your VCN to your on-premises network can use any tunnel that is "up" on your device. Configure your firewalls accordingly. Otherwise, ping tests or application traffic across the connection will not reliably work.

Parameters from API or Console

Get the following parameters from the Oracle Cloud Infrastructure Console or API.

${ipAddress#}

- Oracle VPN headend IPSec tunnel endpoints. There is one value for each tunnel.
- Example value: 129.146.12.52

${sharedSecret#}

- The IPSec IKE pre-shared-key. There is one value for each tunnel.
- Example value: EXAMPLEDPfAMkD7nTH3SWr6OFabdT6exXn6enSlsKbE

${cpePublicIpAddress}

- The public IP address for the CPE (previously made available to Oracle via the Console).

${VcnCidrBlock}

- When creating the VCN, your company selected this CIDR to represent the IP aggregate network for all VCN hosts.
- Example Value: 10.0.0.0/20

Parameters Based on Current CPE Configuration and State

The following parameters need to be discovered by examining your CPE's current configuration and finding valid parameter values that do not overlap with the current CPE configuration.

${cpePublicInterface}

- The name of the FortiGate interface where the CPE IP address is configured.
- Example value: ge-0/0/1.0

${policyId#}

- Index integers, must be unique per rule.
- Requires two per IPSecConnection tunnel.
- Example below for two tunnels requires four unique policyId numbers.
- Example value: 101

Additional Configuration Parameters

The Fortinet config requires the following additional variables:

${vcnID}

- A UUID string used to uniquely name some access-lists and object-groups (can also use any other string that does not create a name that conflicts with an existing object-group or access-list).
- Example: oracle-vcn-1

${VcnCidrNetwork}** and **${VcnCidrNetmask}

- These are the base address and netmask of your VCN Cidr Block
- For more information, see: Wikipedia reference for finding CidrNetmask
- Values based on the example `${VcnCidrBlock}` shown above:
  - `${VcnCidrNetwork}`: 10.0.0.0
  - `${VcnCidrNetmask}`: 255.255.0.0

Config Template Parameter Summary

Each region has multiple Oracle IPSec headends. The template below will allow setting up multiple tunnels on your CPE, each to a corresponding headend. In the table below, "User" is you/your company.

Parameter	Source	Example Value
`${ipAddress1}`	Console/API	129.146.12.52
`${sharedSecret1}`	Console/API	(long string)
`${ipAddress2}`	Console/API	129.146.13.52
`${sharedSecret2}`	Console/API	(long string)
`${cpePublicInterface}`	User CPE	port1
`${policyId1}`	User CPE	101
`${policyId2}`	User CPE	102
`${policyId3}`	User CPE	103
`${policyId4}`	User CPE	104
`${VcnId}`	User/Console	myVCN1
`${VcnCidrNetwork}`	User	10.0.0.0
`${VcnCidrNetmask}`	User	255.255.0.0

ISAKMP Policy Options

Parameter	Recommended Value
ISAKMP protocol version	Version 1
Exchange type	Main mode

Parameter	Recommended Value
Authentication method	Pre-shared keys
Encryption	AES-256-cbc
Authentication algorithm	SHA-384
Diffie-Hellman Group	Group 5
IKE session key lifetime	28800 seconds (8 hours)

IPSec Policy Options

Parameter	Recommended Value
IPSec protocol	ESP, tunnel-mode
Encryption	AES-256-cbc
Authentication algorithm	HMAC-SHA1-96
Diffie-Hellman Group	Group 5
Perfect Forward Secrecy	Enabled
IPSec session key lifetime	3600 seconds (1 hour)

Security Parameter Index

The values for the Security Parameter Index (SPI) depend on whether your CPE supports route-based tunnels or policy-based tunnels. For more information about the correct SPI values to use, see Route-Based Versus Policy-Based IPSec.

CPE Configuration

FIREWALL CONFIGURATION

The configuration example below would allow all traffic from your VCN to any host on your network.

```
config firewall address
    edit any_ipv4
    next
    edit OracleVcn-${VcnId}_remote_subnet
        set subnet ${VcnCidrNetwork} ${VcnCidrNetmask}
    next
end

config firewall addrgrp
    edit OracleVcn-${VcnId}_local
        set member any_ipv4
    next
    edit OracleVcn-${VcnId}_remote
        set member OracleVcn-${VcnId}_remote_subnet
    next
end
config firewall policy
    edit ${policyId1}
        set name vpn_${ipAddress1}_local
        set srcintf ${cpePublicInterface}
        set dstintf ${ipAddress1}
        set srcaddr OracleVcn-${VcnId}_local
        set dstaddr OracleVcn-${VcnId}_remote
        set action accept
        set schedule always
        set service ALL
        set comments "VPN: Oracle ${ipAddress1}"
    next
    edit ${policyId2}
        set name vpn_${ipAddress1}_remote
        set srcintf {ipAddress1}
        set dstintf ${cpePublicInterface}
        set srcaddr OracleVcn-${VcnId}_remote
        set dstaddr OracleVcn-${VcnId}_local
        set action accept
```

```
        set schedule always
        set service ALL
        set comments "VPN: Oracle ${ipAddress1}"
    next
    edit ${policyId3}
        set name vpn_${ipAddress2}_local
        set srcintf ${cpePublicInterface}
        set dstintf ${ipAddress2}
        set srcaddr OracleVcn-${VcnId}_local
        set dstaddr OracleVcn-${VcnId}_remote
        set action accept
        set schedule always
        set service ALL
        set comments "VPN: Oracle ${ipAddress2}"
    next
    edit ${policyId4}
        set name vpn_${ipAddress2}_remote
        set srcintf ${ipAddress2}
        set dstintf ${cpePublicInterface}
        set srcaddr OracleVcn-${VcnId}_remote
        set dstaddr OracleVcn-${VcnId}_local
        set action accept
        set schedule always
        set service ALL
        set comments "VPN: Oracle ${ipAddress2}"
    next
end
```

CONFIGURE ISAKMP PHASE 1

```
config vpn ipsec phase1-interface
    edit ${ipAddress1}
        set interface ${cpePublicInterface}
        set keylife 28800
        set proposal aes256-sha384 aes256-sha256
        set comments "VPN: Oracle ${ipAddress1}"
        set dhgrp 5
        set remote-gw ${ipAddress1}
        set psksecret ${sharedSecret1}
    next
    edit ${ipAddress2}
        set interface ${cpePublicInterface}
        set keylife 28800
```

```
        set proposal aes256-sha384 aes256-sha256
        set comments "VPN: Oracle ${ipAddress2}"
        set dhgrp 5
        set remote-gw ${ipAddress2}
        set psksecret ${sharedSecret2}
    next
end
```

CONFIGURE IPSEC - ISAKMP PHASE 2

```
config vpn ipsec phase2-interface
    edit ${ipAddress1}
        set phase1name ${ipAddress1}
        set proposal aes256-sha1
        set dhgrp 5
        set replay disable
        set auto-negotiate enable
        set comments "VPN: Oracle ${ipAddress1}"
        set keylifeseconds 3600
    next
    edit ${ipAddress2}
        set phase1name ${ipAddress2}
        set proposal aes256-sha1
        set dhgrp 5
        set replay disable
        set auto-negotiate enable
        set comments "VPN: Oracle ${ipAddress2}"
        set keylifeseconds 3600
    next
end
```

CONFIGURE STATIC ROUTES TO YOUR VCN

```
config router static
    edit 1
        set dst ${VcnCidrNetwork} ${VcnCidrNetmask}
        set device ${ipAddress1}
        set comment "Oracle VPN-${ipAddress1}"
    next
    edit 2
        set dst ${VcnCidrNetwork} ${VcnCidrNetmask}
        set device ${ipAddress2}
        set comment "Oracle VPN-${ipAddress2}"
```

```
    next
end
```

Juniper MX

This configuration was validated using a Juniper MX 240 running Junos 15.1.



Important

Oracle uses asymmetric routing across the multiple tunnels that make up the IPSec VPN connection. Even if you configure one tunnel as primary and another as backup, traffic from your VCN to your on-premises network can use any tunnel that is "up" on your device. Configure your firewalls accordingly. Otherwise, ping tests or application traffic across the connection will not reliably work.

Parameters from API or Console

Get the following parameters from the Oracle Cloud Infrastructure Console or API.

${ipAddress#}

- Oracle VPN headend IPSec tunnel endpoints. There is one value for each tunnel.
- Example value: 129.146.12.52

${sharedSecret#}

- The IPSec IKE pre-shared-key. There is one value for each tunnel.
- Example value: EXAMPLEDPfAMkD7nTH3SWr6OFabdT6exXn6enSlsKbE

${cpePublicIpAddress}

- The public IP address for the CPE (previously made available to Oracle via the Console).

${VcnCidrBlock}

- When creating the VCN, your company selected this CIDR to represent the IP aggregate network for all VCN hosts.
- Example Value: 10.0.0.0/20

Parameters Based on Current CPE Configuration and State

The following parameters need to be discovered by examining your CPE's current configuration and finding valid parameter values that do not overlap with the current CPE configuration.

${cpePublicInterface}

- The name of the Juniper interface where the CPE IP address is configured.
- Example value: ge-0/0/1.0

${msInterface#} - one per tunnel

- These interfaces correspond to one of the four encryption ASICs on the MS-MPC card.
- You can distribute load across the ASICs by spreading your tunnels across them.
- Example values: ms-2/3/0, ms-2/3/1

${insideMsUnit1}** and **${outsideMsUnit1} - one pair per tunnel

- For every tunnel, you will need an ms-mpc interface pair of units.
- One represents the outside of the IPSec tunnel.
- The other the inside of the tunnel.
- The router forwards packets from your on-premises network to your VCN into the inside unit.
    - The encryption ASIC then encrypts the packets based on the rules and policies.
    - Then, the encrypted packet egresses out the outside unit as an ESP packet, ready to be forwarded to Oracle's VPN headend routers.

- There are a little over 16,000 possible values for unit numbers.
    - One way to allocate the units is to offset them by 8,000.
    - You can pick values between 0 - 7999 for insideMsUnits and 8000-15999 for outsideMsUnits.

Config Template Parameter Summary

Each region has multiple Oracle IPSec headends. The template below will allow setting up multiple tunnels on your CPE, each to a corresponding headend. In the table below, "User" is you/your company.

Parameter	Source	Example Value
`${ipAddress1}`	Console/API	129.146.12.52
`${sharedSecret1}`	Console/API	(long string)
`${ipAddress2}`	Console/API	129.146.13.52
`${sharedSecret2}`	Console/API	(long string)
`${cpePublicIpAddress}`	User	1.2.3.4
`${msInterface1}`	User	ms-2/3/0
`${msInterface2}`	User	ms-2/3/1
`${insideMsUnit1}`	User	101
`${insideMsUnit2}`	User	102
`${outsideMsUnit1}`	User	8101
`${outsideMsUnit2}`	User	8102
`${VcnCidrBlock}`	User	10.0.0.0/16

ISAKMP Policy Options

Parameter	Recommended Value
ISAKMP protocol version	Version 1
Exchange type	Main mode
Authentication method	Pre-shared keys
Encryption	AES-256-cbc
Authentication algorithm	SHA-384
Diffie-Hellman Group	Group 5
IKE session key lifetime	28800 seconds (8 hours)

IPSec Policy Options

Parameter	Recommended Value
IPSec protocol	ESP, tunnel-mode
Encryption	AES-256-cbc
Authentication algorithm	HMAC-SHA1-96
Diffie-Hellman Group	Group 5
Perfect Forward Secrecy	Enabled
IPSec session key lifetime	3600 seconds (1 hour)

Security Parameter Index

The values for the Security Parameter Index (SPI) depend on whether your CPE supports route-based tunnels or policy-based tunnels. For more information about the correct SPI values to use, see Route-Based Versus Policy-Based IPSec.

CPE Configuration

Note on routing-instances: If you are using routing-instances on your CPE, you need to make sure you account for them in your configuration. The main configuration template below does not account for routing-instances, so you would need to merge the following config excerpt into the non-routing-instance templates.

```
routing-instances {
    ${customer premise routing instance} {
        interface ${msInterface1}.${insideMsUnit1};
        interface ${msInterface2}.${insideMsUnit2};
        routing-options {
            static {
                route ${vcnCidrBlock} next-hop [ ${msInterface1}.${insideMsUnit1}
${msInterface2}.${insideMsUnit2} ]
            }
        }
    }
    ${internet-routing-instance} {
        interface ${msInterface1}.${insideMsUnit1};
        interface ${msInterface2}.${insideMsUnit2};
    }
}
services {
    service-set oracle-vpn-tunnel-${ipAddress1} {
        ipsec-vpn-options {
            local-gateway ${cpePublicIpAddress} routing-instance ${internet-routing-instance};
        }
    }
    service-set oracle-vpn-tunnel-${ipAddress2} {
        ipsec-vpn-options {
            local-gateway ${cpePublicIpAddress} routing-instance ${internet-routing-instance};
        }
    }
    service-set oracle-vpn-tunnel-${ipAddress2} {
        ipsec-vpn-options {
            local-gateway ${cpePublicIpAddress} routing-instance ${internet-routing-instance};
        }
    }
}
```

TUNNEL INTERFACE CONFIGURATION

This configures the Juniper MX interfaces that represent the "inside" and "outside" of the IPSec tunnels. There is one pair of interface/units per tunnel. The unit numbers must be unique on your router.

```
interfaces {
    ${msInterface1} {
        unit ${insideMsUnit1} {
            description oracle-vpn-tunnel-${ipAddress1}-INSIDE;
            family inet;
            service-domain inside;
        }
        unit ${outsideMsUnit1} {
            description oracle-vpn-tunnel-${ipAddress1}-OUTSIDE;
            family inet;
            service-domain outside;
        }
    }
    ${msInterface2} {
        unit ${insideMsUnit2} {
            description oracle-vpn-tunnel-${ipAddress2}-INSIDE;
            family inet;
            service-domain inside;
        }
        unit ${outsideMsUnit2} {
            description oracle-vpn-tunnel-${ipAddress2}-OUTSIDE;
            family inet;
            service-domain outside;
        }
    }
}
```

CPE TO VCN STATIC ROUTES

The IPSec tunnels require static routes to get traffic to go through them. The routes are configured to point down the tunnel interfaces. If the IPSec tunnel is down, the Juniper MX will stop using that route. You should redistribute these routes into your customer premise network.

```
routing-options {
    static {
```

```
        route ${vcnCidrBlock} next-hop [ ${msInterface1}.${insideMsUnit1}
${msInterface2}.${insideMsUnit2} ]
    }
}
```

SERVICES CONFIGURATION

The IPSec tunnels are configured in this section.

```
services {
    service-set oracle-vpn-tunnel-${ipAddress1} {
        next-hop-service {
            inside-service-interface ${msInterface1}.${insideMsUnit1};
            outside-service-interface ${msInterface1}.${outsideMsUnit1};
        }
        ipsec-vpn-options {
            local-gateway ${cpePublicIpAddress}
        }
        ipsec-vpn-rules oracle-vpn-tunnel-${ipAddress1};
    }
    service-set oracle-vpn-tunnel-${ipAddress2} {
        next-hop-service {
            inside-service-interface ${msInterface2}.${insideMsUnit2};
            outside-service-interface ${msInterface2}.${outsideMsUnit2};
        }
        ipsec-vpn-options {
            local-gateway ${cpePublicIpAddress}
        }
        ipsec-vpn-rules oracle-vpn-tunnel-${ipAddress2};
    }
    ipsec-vpn {
        rule oracle-vpn-tunnel-${ipAddress1} {
            term 1 {
                from {
                    ipsec-inside-interface ${msInterface1}.${insideMsUnit1};
                }
                then {
                    remote-gateway ${ipAddress1};
                    dynamic {
                        ike-policy ike-policy-${ipAddress1};
                        ipsec-policy oracle-ike-policy;
                    }
                    tunnel-mtu 1420;
```

```
                initiate-dead-peer-detection;
                dead-peer-detection {
                    interval 5;
                    threshold 4;
                }
            }
        }
        match-direction input;
    }
    rule oracle-vpn-tunnel-${ipAddress2} {
        term 1 {
            from {
                ipsec-inside-interface ${msInterface2}.${insideMsUnit2};
            }
            then {
                remote-gateway ${ipAddress2};
                dynamic {
                    ike-policy ike-policy-${ipAddress2};
                    ipsec-policy oracle-ike-policy;
                }
                tunnel-mtu 1420;
                initiate-dead-peer-detection;
                dead-peer-detection {
                    interval 5;
                    threshold 4;
                }
            }
        }
        match-direction input;
    }
    ipsec {
        proposal esp-aes256-sha1 {
            protocol esp;
            authentication-algorithm hmac-sha1-96;
            encryption-algorithm aes-256-cbc;
            lifetime-seconds 3600;
        }
        policy oracle-ipsec-policy {
            perfect-forward-secrecy {
                keys group5;
            }
            proposals [ esp-aes256-sha1 ];
```

```
            }
        }
        ike {
            proposal aes256-sha384-group5 {
                authentication-method pre-shared-keys;
                dh-group group5;
                authentication-algorithm sha-384;
                encryption-algorithm aes-256-cbc;
                lifetime-seconds 28800;
            }
            policy oracle-ike-policy-${ipAddress1} {
                mode main;
                version 1;
                proposals [ aes256-sha384-group5 ];
                local-id ipv4_addr ${cpePublicIpAddress};
                remote-id ipv4_addr ${ipAddress1};
                pre-shared-key ascii-text ${sharedSecret1};
            }
            policy oracle-ike-policy-${ipAddress2} {
                mode main;
                version 1;
                proposals [ aes256-sha384-group5 ];
                local-id ipv4_addr ${cpePublicIpAddress};
                remote-id ipv4_addr ${ipAddress2};
                pre-shared-key ascii-text ${sharedSecret2};

            }
        }
    }
}
```

Juniper SRX

This configuration was validated using a Juniper SRX 240 running 12.1X44-D35.5.



Important

Oracle uses asymmetric routing across the multiple tunnels that make up the IPSec VPN connection. Even if you configure one tunnel as primary and another as backup, traffic from your VCN to your on-premises network can use any tunnel that is "up" on your device. Configure your firewalls accordingly. Otherwise, ping tests or application traffic across the connection will not reliably work.

Parameters from API or Console

Get the following parameters from the Oracle Cloud Infrastructure Console or API.

${ipAddress#}

- Oracle VPN headend IPSec tunnel endpoints. There is one value for each tunnel.
- Example value: 129.146.12.52

${sharedSecret#}

- The IPSec IKE pre-shared-key. There is one value for each tunnel.
- Example value: EXAMPLEDPfAMkD7nTH3SWr6OFabdT6exXn6enSlsKbE

${cpePublicIpAddress}

- The public IP address for the CPE (previously made available to Oracle via the Console).

${VcnCidrBlock}

- When creating the VCN, your company selected this CIDR to represent the IP aggregate network for all VCN hosts.
- Example Value: 10.0.0.0/20

Parameters Based on Current CPE Configuration and State

The following parameters need to be discovered by examining your CPE's current configuration and finding valid parameter values that do not overlap with the current CPE configuration.

${cpePublicInterface}

- The name of the Juniper interface where the CPE IP address is configured.
- Example value: ge-0/0/1.0

${tunnelUnit#} - one per tunnel

- You will need multiple tunnel unit numbers per IPSec connection.
- One per IPSec tunnel.
- Oracle recommends setting up all Oracle configured tunnels for maximum redundancy.
- The Juniper SRX uses the `st0` interface for IPSec tunnels.
- Example values: 1, 2

Juniper Security Zones

- Juniper SRX has the concept of "security zones".
- You will need two different security zones for your VPN.

${InsideZoneName}

- This zone contains the interfaces that are part of your internal network that need to reach resources in your Oracle VCN.

${OracleVpnZoneName}

- This zone contains the interfaces that are part of the Oracle Cloud Infrastructure network.
- This includes the inside of the IPSec tunnel interfaces.

${InternetSecurityZoneName}

- This is the zone that includes your `${cpePublicInterface}`.

Config Template Parameter Summary

Each region has multiple Oracle IPSec headends. The template below will allow setting up multiple tunnels on your CPE, each to a corresponding headend. In the table below, "User" is you/your company.

Parameter	Source	Example Value
`${ipAddress1}`	Console/API	129.146.12.52
`${sharedSecret1}`	Console/API	(long string)
`${ipAddress2}`	Console/API	129.146.13.52
`${sharedSecret2}`	Console/API	(long string)
`${cpePublicInterface}`	User	ge-0/0/1.0
`${tunnelUnit1}`	User	1
`${tunnelUnit2}`	User	2
`${OracleVpnZoneName}`	User	oracle-vpn
`${InternetSecurityZoneName}`	User	INTERNET
`${VcnCidrBlock}`	User	10.0.0.0/16

ISAKMP Policy Options

Parameter	Recommended Value
ISAKMP protocol version	Version 1
Exchange type	Main mode

Parameter	Recommended Value
Authentication method	Pre-shared keys
Encryption	AES-256-cbc
Authentication algorithm	SHA-384
Diffie-Hellman Group	Group 5
IKE session key lifetime	28800 seconds (8 hours)

IPSec Policy Options

Parameter	Recommended Value
IPSec protocol	ESP, tunnel-mode
Encryption	AES-256-cbc
Authentication algorithm	HMAC-SHA1-96
Diffie-Hellman Group	Group 5
Perfect Forward Secrecy	Enabled
IPSec session key lifetime	3600 seconds (1 hour)

Security Parameter Index

The values for the Security Parameter Index (SPI) depend on whether your CPE supports route-based tunnels or policy-based tunnels. For more information about the correct SPI values to use, see Route-Based Versus Policy-Based IPSec.

CPE Configuration

TUNNEL INTERFACE CONFIGURATION

This configures the Juniper SRX interfaces that represent the "inside" of the IPSec tunnels. There is one interface/unit per tunnel.

```
interfaces {
    st0 {
        unit ${tunnelUnit1} {
            family inet;
        }
        unit ${tunnelUnit2} {
            family inet;
        }
    }
}
```

CPE TO VCN STATIC ROUTES

The IPSec tunnels require static routes to get traffic to go through them. The routes are configured to point down the tunnel interfaces. If the IPSec tunnel is down, the Juniper SRX will stop using that route. You should redistribute these routes into your customer premise network.

```
routing-options {
    static {
        route ${VcnCidrBlock} next-hop [ st0.${tunnelUnit1} st0.${tunnelUnit2} ];
    }
}
```

SECURITY POLICY

The security policy consists of four sections:

- ike: IKE policy parameters.
- ipsec: IPSec policy parameters.
- zones: Defines Juniper zones and place interfaces into logical zones.
- policies: Defines what traffic is permitted in and out of and between zones.

```
security {
    ike {
        proposal oracle-ike-proposal {
            authentication-method pre-shared-keys;
            dh-group group5;
            authentication-algorithm sha-384;
            encryption-algorithm aes-256-cbc;
            lifetime-seconds 28800;
        }
        policy ike_pol_oracle-vpn-${ipAddress1} {
            mode main;
            proposals oracle-ike-proposal;
            pre-shared-key ascii-text ${sharedSecret1}
        }
        policy ike_pol_oracle-vpn-${ipAddress2} {
            mode main;
            proposals oracle-ike-proposal;
            pre-shared-key ascii-text ${sharedSecret2}
        }
        gateway gw_oracle-${ipAddress1} {
            ike-policy ike_pol_oracle-vpn-${ipAddress1};
            address ${ipAddress1};
            dead-peer-detection;
            external-interface ${cpePublicInterface};
        }
        gateway gw_oracle-${ipAddress2} {
            ike-policy ike_pol_oracle-vpn-${ipAddress2};
            address ${ipAddress2};
            dead-peer-detection;
            external-interface ${cpePublicInterface};
        }
    }
    ipsec {
        vpn-monitor-options;
        proposal oracle-ipsec-proposal {
            protocol esp;
            authentication-algorithm hmac-sha1-96;
            encryption-algorithm aes-256-cbc;
            lifetime-seconds 3600;
        }
        policy ipsec_pol_oracle-vpn {
            perfect-forward-secrecy {
```

```
            keys group5;
        }
        proposals oracle-ipsec-proposal;
    }
    vpn oracle-vpn-${ipAddress1} {
        bind-interface st0.${tunnelUnit1};
        vpn-monitor;
        ike {
            gateway gw_oracle-${ipAddress1};
            ipsec-policy ipsec_pol_oracle-vpn;
        }
        establish-tunnels immediately;
    }
    vpn oracle-vpn-${ipAddress2} {
        bind-interface st0.${tunnelUnit2};
        vpn-monitor;
        ike {
            gateway gw_oracle-${ipAddress2};
            ipsec-policy ipsec_pol_oracle-vpn;
        }
        establish-tunnels immediately;
    }
}
policies {
    from-zone ${InsideZoneName} to-zone ${OracleVpnZoneName} {
        policy new_vpn-out {
            match {
                source-address any-ipv4;
                destination-address any-ipv4;
                application any;
                source-identity any;
            }
            then {
                permit;
            }
        }
        policy vpn-out {
            match {
                source-address any-ipv4;
                destination-address any-ipv4;
                application any;
                source-identity any;
```

```
                }
                then {
                    permit;
                }
            }
        }
    }
    zones {
        security-zone ${OracleVpnZoneName} {
            interfaces {
                st0.${tunnelUnit1} {
                    host-inbound-traffic {
                        protocols {
                            bgp;
                        }
                    }
                }
                st0.${tunnelUnit2} {
                    host-inbound-traffic {
                        protocols {
                            bgp;
                        }
                    }
                }
            }
        }
        security-zone ${InternetSecurityZoneName} {
            interfaces {
                ${cpePublicInterface} {
                    host-inbound-traffic {
                        system-services {
                            ike;
                            ping;
                        }
                    }
                }
            }
        }
    }
}
```

Libreswan

Libreswan is an open-source IPSec implementation that is based on FreeS/WAN and Openswan. Most Linux distributions include Libreswan or make it easy to install. You can install it on hosts in either your on-premises network or a cloud provider network. For an example of setting up a Libreswan host in another cloud provider to connect to your Oracle Cloud Infrastructure VCN, see Access to Other Clouds with Libreswan.

This configuration was validated using Libreswan version 3.23-4.



Important

Oracle uses asymmetric routing across the multiple tunnels that make up the IPSec VPN connection. Even if you configure one tunnel as primary and another as backup, traffic from your VCN to your on-premises network can use any tunnel that is "up" on your device. Configure your firewalls accordingly. Otherwise, ping tests or application traffic across the connection will not reliably work.

Route-Based VPNs with Libreswan

Libreswan supports both route-based and policy-based tunnels. The tunnel types can coexist without interfering with each other. The Oracle VPN headends use route-based tunnels. Oracle recommends that you configure Libreswan with the Virtual Tunnel Interface (VTI) configuration syntax.

ISAKMP and IPSec Policy Options

Libreswan's default values for ISAKMP and IPSec policy options are compatible with Oracle's VPN headends. For more information about those policy options, see Generic CPE Configuration Information.

Default Libreswan Configuration Files

Libreswan configuration uses the concept of *left* and *right* to define the configuration parameters for your local CPE device and the remote gateway. Either side of the connection (the *conn* in the Libreswan configuration) can be left or right, but the configuration for that connection must be consistent. In this example:

- **left:** your local Libreswan CPE
- **right:** the Oracle VPN gateway

/ETC/IPSEC.CONF AND /ETC/IPSEC.SECRETS

The default Libreswan installation creates the following files:

- `etc/ipsec.conf`: The root of the Libreswan configuration.
- `/etc/ipsec.secrets`: The root of the location where Libreswan looks for secrets (the tunnel preshared keys).
- `/etc/ipsec.d/`: A directory for storing the `.conf` and `.secrets` files for your Oracle Cloud Infrastructure tunnels (for example: `oci-ipsec.conf` and `oci-ipsec.secrets`). Libreswan encourages you to create these files in this folder.

The default `etc/ipsec.conf` file includes this line:

```
include /etc/ipsec.d/*.conf
```

The default `etc/ipsec.secrets` file includes this line:

```
include /etc/ipsec.d/*.secrets
```

These lines automatically merge all the `.conf` and `.secrets` files in the `/etc/ipsec.d` directory into the main configuration and secrets files that Libreswan uses.



Important

Do not change the default `.conf` and `.secrets` files unless you're experienced with Libreswan and know the implications.

Parameters from API or Console

Get the following parameters from the Oracle Cloud Infrastructure Console or API.

${ipAddress#}

- Oracle VPN headend IPSec tunnel endpoints. There is one value for each tunnel.
- Example value: 129.146.12.52

${psk#}

- The IPSec IKE pre-shared-key. There is one value for each tunnel.
- Example value: EXAMPLEDPfAMkD7nTH3SWr6OFabdT6exXn6enSlsKbE

${cpePublicIpAddress}

- The public IP address for the CPE (previously made available to Oracle via the Console).
- Example value: 1.2.3.4

${VcnCidrBlock}

- When creating the VCN, your company selected this CIDR to represent the IP aggregate network for all VCN hosts.
- Example value: 10.0.0.0/20

Additional Configuration Parameters

${cpeLocalIP}

- The IP address configured directly on your Libreswan host
- If your CPE is not behind a 1-1 NAT, this value is the same as `${cpePublicIpAddress}`
- Example value:
  - If your CPE is not behind a 1-1 NAT: 1.2.3.4
  - If your CPE is behind a 1-1 NAT: 10.2.3.4

${vti#}

- A name of your choice for the virtual tunnel interface. There is one value for each tunnel.
- Example value: vti01

${markValue#}

- These values are used to mark packets into the Libreswan host that match a particular virtual tunnel interface (vti).
- Each mark value must be unique across the Libreswan host.
- If the mark value is changed after the vti interface is created, you must remove and recreate the vti interface (see the procedure that follows).
- Example value for each tunnel: 5/0xffffffff and 6/0xffffffff

To recreate the vti interfaces after changing mark values

If you change the mark values in the Libreswan config, you must remove and recreate the vti interfaces.

For example, if the vti interface is named vti01, use this command:

```
sudo ip link delete vti01
```

To verify the interface has been deleted, use this command:

```
ip link show
```

Make sure to restart the Libreswan service (also see Reloading the Libreswan Configuration):

```
service ipsec restart
```

Config Template Parameter Summary

Each region has multiple Oracle IPSec headends. The template that follows lets you set up multiple tunnels on your CPE, each to a corresponding headend. In the following table, "User" is you or your company.

Parameter	Source	Example Value
`${ipAddress1}`	Console/API	129.146.12.52
`${psk1}`	Console/API	(long string)
`${ipAddress2}`	Console/API	129.146.13.52
`${psk2}`	Console/API	(long string)
`${VcnCidrBlock}`	User	10.0.0.0/20
`${cpePublicIpAddress}`	User	1.2.3.4
`${cpeLocalIP}`	User	If your CPE is not behind a 1-1 NAT: 1.2.3.4 If your CPE is behind a 1-1 NAT: 10.2.3.4
`${vti1}`	User	vti01
`${vti2}`	User	vti02
`${markValue1}`	User	5/0xffffffff
`${markValue2}`	User	6/0xffffffff

Setting Up Your Configuration File: /etc/ipsec.d/oci-ipsec.conf

Use the following template for your `/etc/ipsec.d/oci-ipsec.conf` file. The file defines the two tunnels that Oracle creates when you set up the IPSec connection.



Important

If your CPE is behind a 1-1 NAT device, uncomment the `leftid` parameter and set it equal to the `${cpePublicIpAddress}`.

```
conn oracle-tunnel-1
     left=${cpeLocalIP}
     # leftid=${cpePublicIpAddress} # See preceding note about 1-1 NAT device
     right=${ipAddress1}
     authby=secret
     leftsubnet=0.0.0.0/0
     rightsubnet=0.0.0.0/0
     auto=start
     mark=${markValue1}
     vti-interface=${vti1}
     vti-routing=no
     encapsulation=no
     ikelifetime=28800s
     salifetime=3600s
conn oracle-tunnel-2
     left=${cpeLocalIP}
     # leftid=${cpePublicIpAddress} # See preceding note about 1-1 NAT device
     right=${ipAddress2}
     authby=secret
     leftsubnet=0.0.0.0/0
     rightsubnet=0.0.0.0/0
     auto=start
     mark=${markValue2}
     vti-interface=${vti2}
     vti-routing=no
     encapsulation=no
     ikelifetime=28800s
     salifetime=3600s
```

Setting Up Your Secrets File: /etc/ipsec.d/oci-ipsec.secrets

Use the following template for your `/etc/ipsec.d/oci-ipsec.secrets` file. It contains two lines per IPSec connection (one line per tunnel).

```
${cpePublicIpAddress} ${ipAddress1}: PSK "${psk1}"
${cpePublicIpAddress} ${ipAddress2}: PSK "${psk2}"
```

Reloading the Libreswan Configuration

After setting up your configuration and secrets files, you must restart the Libreswan service.



Important

Restarting the Libreswan service may impact existing tunnels.

The following command rereads the config file and restarts the Libreswan service. If you're logged in with an unprivileged user account, you might need to use `sudo` before the command.

```
service ipsec restart
```

Checking the Libreswan Status

Check the current state of your Libreswan tunnels by using the following command.

```
ipsec status
```

The tunnel is established if you see a line that includes the following:

```
STATE_MAIN_I4: ISAKMP SA established
```

In the future, if you need to open a support ticket with Oracle about your Libreswan tunnel, include the output of the preceding `ipsec status` command.

Checking the Tunnel Interface Status

Check if the virtual tunnel interfaces are up or down by using the `ifconfig` command or the `ip link show` command. You can also use applications such as tcpdump with the interfaces.

Here's an example of the `ifconfig` output:

```
ifconfig
<output trimmed>

vti01: flags=209<UP,POINTOPOINT,RUNNING,NOARP> mtu 8980
     inet6 fe80::5efe:a00:2 prefixlen 64 scopeid 0x20<link>
     tunnel txqueuelen 1000 (IPIP Tunnel)
     RX packets 0 bytes 0 (0.0 B)
     RX errors 0 dropped 0 overruns 0 frame 0
     TX packets 0 bytes 0 (0.0 B)
     TX errors 10 dropped 0 overruns 0 carrier 10 collisions 0

vti02: flags=209<UP,POINTOPOINT,RUNNING,NOARP> mtu 8980
     inet6 fe80::5efe:a00:2 prefixlen 64 scopeid 0x20<link>
     tunnel txqueuelen 1000 (IPIP Tunnel)
     RX packets 0 bytes 0 (0.0 B)
     RX errors 0 dropped 0 overruns 0 frame 0
     TX packets 0 bytes 0 (0.0 B)
     TX errors 40 dropped 0 overruns 0 carrier 40 collisions 0
```

Here's an example of the `ip link show` output:

```
ip link show
<output trimmed>
```

```
9: vti01@NONE: <POINTOPOINT,NOARP,UP,LOWER_UP> mtu 8980 qdisc noqueue
state UNKNOWN mode DEFAULT group default qlen 1000
   link/ipip 10.0.0.2 peer 129.213.240.52

10: vti02@NONE: <POINTOPOINT,NOARP,UP,LOWER_UP> mtu 8980 qdisc noqueue
state UNKNOWN mode DEFAULT group default qlen 1000
   link/ipip 10.0.0.2 peer 129.213.240.51
```

Configuring IP Routing

Use the following `ip` command to create static routes that send traffic to your VCN through the IPSec tunnels. If you're logged in with an unprivileged user account, you might need to use `sudo` before the command.

```
ip route add ${VcnCidrBlock} nexthop dev ${vti1} nexthop dev ${vti2}
ip route show
```

Openswan

If you want to use Openswan to create an IPSec VPN to Oracle Cloud Infrastructure, see Libreswan.

How Openswan and Libreswan Are Related

Openswan is a well-known IPSec implementation for Linux. It began as a fork of the now-defunct FreeS/WAN project in 2003. Unlike the FreeS/WAN project, it didn't exclusively target the GNU/Linux operation system, but expanded its use to other operating systems. In 2012, FreeS/WAN renamed itself to The Libreswan Project because of a lawsuit over a trademark of the name *openswan*.

As a result, when you try to install or query the Openswan package on Oracle Linux, by default the Libreswan package is installed or shown instead. The following yum search query command illustrates this behavior:

```
$ sudo yum search openswan
Loaded plugins: langpacks, ulninfo
Matched: openswan ============================================
NetworkManager-libreswan.x86_64 : NetworkManager VPN plugin for libreswan
NetworkManager-libreswan-gnome.x86_64 : NetworkManager VPN plugin for libreswan-GNOME files
libreswan.x86_64 : IPsec implementation with IKEv1 and IKEv2 keying protocols
```

Libreswan is maintained by The Libreswan Project and has been under active development for over 15 years, going back to the FreeS/WAN Project. For more information, see the project's history.

Palo Alto

This configuration was validated using a PA-500 running PanOS version 6.0.6.



Important

Oracle uses asymmetric routing across the multiple tunnels that make up the IPSec VPN connection. Even if you configure one tunnel as primary and another as backup, traffic from your VCN to your on-premises network can use any tunnel that is "up" on your device. Configure your firewalls accordingly. Otherwise, ping tests or application traffic across the connection will not reliably work.

Parameters from API or Console

Get the following parameters from the Oracle Cloud Infrastructure Console or API.

${ipAddress#}

- Oracle VPN headend IPSec tunnel endpoints. There is one value for each tunnel.
- Example value: 129.146.12.52

${sharedSecret#}

- The IPSec IKE pre-shared-key. There is one value for each tunnel.
- Example value: EXAMPLEDPfAMkD7nTH3SWr6OFabdT6exXn6enSlsKbE

${cpeInterfaceName}

- The name of the CPE interface where the CPE IP address is configured.
- Example Value: ethernet1/1

${VcnCidrBlock}

- When creating the VCN, your company selected this CIDR to represent the IP aggregate network for all VCN hosts.
- Example Value: 10.0.0.0/20

Parameters Discovered from CPE Configuration

The following parameters are based on your current CPE configuration.

${tunnelUnit#}

- Each tunnel needs a unit number that identifies that tunnel.
- Example: 10, where 10 is the tunnelUnit for interface tunnel.10

${oracleSecurityZoneName}

- The tunnels need to be placed inside a security zone that defines their access profile.
- Example: "Oracle Cloud Infrastructure"
- Note: The value must be enclosed in quotation marks.

${CpeVirtualRouterName}

- The tunnels terminate into a virtual router in the Palo Alto. You can either terminate them into an existing virtual router, or configure a new virtual router.
- Example Value: Oracle-virtual-router

ISAKMP Policy Options

Parameter	Recommended Value
ISAKMP protocol version	Version 1
Exchange type	Main mode
Authentication method	Pre-shared keys
Encryption	AES-256-cbc
Authentication algorithm	SHA-384
Diffie-Hellman Group	Group 5
IKE session key lifetime	28800 seconds (8 hours)

IPSec Policy Options

Parameter	Recommended Value
IPSec protocol	ESP, tunnel-mode
Encryption	AES-256-cbc
Authentication algorithm	HMAC-SHA1-96
Diffie-Hellman Group	Group 5
Perfect Forward Secrecy	Enabled
IPSec session key lifetime	3600 seconds (1 hour)

Security Parameter Index

The values for the Security Parameter Index (SPI) depend on whether your CPE supports route-based tunnels or policy-based tunnels. For more information about the correct SPI values to use, see Route-Based Versus Policy-Based IPSec.

Common ISAKMP and IPSec Profile Configuration

```
set network ike crypto-profiles ike-crypto-profiles oracle-bare-metal-cloud-ike encryption aes256
set network ike crypto-profiles ike-crypto-profiles oracle-bare-metal-cloud-ike hash sha384
set network ike crypto-profiles ike-crypto-profiles oracle-bare-metal-cloud-ike dh-group group5
set network ike crypto-profiles ike-crypto-profiles oracle-bare-metal-cloud-ike lifetime hours 8
set network ike crypto-profiles ipsec-crypto-profiles oracle-bare-metal-cloud-ipsec esp encryption
aes256
set network ike crypto-profiles ipsec-crypto-profiles oracle-bare-metal-cloud-ipsec esp authentication
sha1
set network ike crypto-profiles ipsec-crypto-profiles oracle-bare-metal-cloud-ipsec dh-group group5
set network ike crypto-profiles ipsec-crypto-profiles oracle-bare-metal-cloud-ipsec lifetime hours 1
```

ISAKMP Gateway Configuration

```
set network ike gateway oracle-gateway-${ipAddress1} authentication pre-shared-key key ${sharedSecret1}
set network ike gateway oracle-gateway-${ipAddress1} protocol-common nat-traversal enable no
set network ike gateway oracle-gateway-${ipAddress1} local-address interface ${cpeInterfaceName}
set network ike gateway oracle-gateway-${ipAddress1} peer-address ip ${ipAddress1}
set network ike gateway oracle-gateway-${ipAddress1} protocol ikev1 ike-crypto-profile oracle-bare-
metal-cloud-ike

set network ike gateway oracle-gateway-${ipAddress2} authentication pre-shared-key key ${sharedSecret2}
set network ike gateway oracle-gateway-${ipAddress2} protocol-common nat-traversal enable no
set network ike gateway oracle-gateway-${ipAddress2} local-address interface ${cpeInterfaceName}
set network ike gateway oracle-gateway-${ipAddress2} peer-address ip ${ipAddress2}
set network ike gateway oracle-gateway-${ipAddress2} protocol ikev1 ike-crypto-profile oracle-bare-
metal-cloud-ike
```

IPSec Configuration

```
set network tunnel ipsec oracle-ipsec-vpn-${ipAddress1} auto-key ike-gateway oracle-gateway-
${ipAddress1}
set network tunnel ipsec oracle-ipsec-vpn-${ipAddress1} auto-key ipsec-crypto-profile oracle-bare-metal-
cloud-ipsec
set network tunnel ipsec oracle-ipsec-vpn-${ipAddress1} tunnel-monitor enable no
set network tunnel ipsec oracle-ipsec-vpn-${ipAddress1} tunnel-interface tunnel.${tunnelUnit1}

set network tunnel ipsec oracle-ipsec-vpn-${ipAddress2} auto-key ike-gateway oracle-gateway-
${ipAddress2}
set network tunnel ipsec oracle-ipsec-vpn-${ipAddress2} auto-key ipsec-crypto-profile oracle-bare-metal-
cloud-ipsec
```

```
set network tunnel ipsec oracle-ipsec-vpn-${ipAddress2} tunnel-monitor enable no
set network tunnel ipsec oracle-ipsec-vpn-${ipAddress2} tunnel-interface tunnel.${tunnelUnit2}
```

Virtual Router Configuration

Newer versions of PanOS support Equal-cost multipath routing (ECMP).

```
set network virtual-router ${CpeVirtualRouterName} interface [ tunnel.101 tunnel.102 ]
set network virtual-router ${CpeVirtualRouterName} routing-table ip static-route oracle-vpn-
${ipAddress1} destination ${VcnCidrBlock}
set network virtual-router ${CpeVirtualRouterName} routing-table ip static-route oracle-vpn-
${ipAddress1} interface tunnel.${tunnelUnit1}
set network virtual-router ${CpeVirtualRouterName} routing-table ip static-route oracle-vpn-
${ipAddress1} metric 10
set network virtual-router ${CpeVirtualRouterName} routing-table ip static-route oracle-vpn-
${ipAddress2} destination ${VcnCidrBlock}
set network virtual-router ${CpeVirtualRouterName} routing-table ip static-route oracle-vpn-
${ipAddress2} interface tunnel.${tunnelUnit2}
set network virtual-router ${CpeVirtualRouterName} routing-table ip static-route oracle-vpn-
${ipAddress2} metric 11
```

Security Zone Configuration

```
set zone ${oracleSecurityZoneName} network layer3 [ tunnel.101 tunnel.102 ]
```

WatchGuard

You can configure a WatchGuard Firebox as your CPE device for an IPSec VPN.

Go to the WatchGuard knowledge base article to download the configuration instructions.

Yamaha RTX Series

This configuration was validated using a RTX1210 running Firmware Rev.14.01.28 and RTX830 running Firmware Rev.15.02.03.



Important

Oracle uses asymmetric routing across the multiple tunnels that make up the IPSec VPN connection. Even if you configure one tunnel as primary and another as backup, traffic from your VCN to your on-premises network can use any tunnel that is "up" on your device. Configure your firewalls accordingly. Otherwise, ping tests or application traffic across the connection will not reliably work.

Before Starting

Before configuring your CPE, make sure to:

- Configure your internet provider settings
- Configure firewall rules to open UDP port 500, UDP port 4500, and ESP.

Parameters from API or Console

Get the following parameters from the Oracle Cloud Infrastructure Console or API.

${ipAddress#}

- Oracle VPN headend IPSec tunnel endpoints. There is one value for each tunnel.
- Example value: 129.146.12.52

${sharedSecret#}

- The IPSec IKE pre-shared-key. There is one value for each tunnel.
- Example value: EXAMPLEDPfAMkD7nTH3SWr6OFabdT6exXn6enSlsKbE

${cpePublicIpAddress}

- The public IP address for the CPE (previously made available to Oracle via the Console).

${VcnCidrBlock}

- When creating the VCN, your company selected this CIDR to represent the IP aggregate network for all VCN hosts.
- Example Value: 10.0.0.0/20

Parameters Based on Current CPE Configuration and State

The following parameters are based on your current CPE configuration.

${tunnelInterface#}

- An interface number to identify the specific tunnel.
- Example value: 1

${ipsecPolicy#}

- The SA policy to be used for the selected inline interface.
- Example value: 1

${localAddress}

- The public IP address of your CPE.
- Example value: 146.56.2.52

Config Template Parameter Summary

Each region has multiple Oracle IPSec headends. The template below allows you to set up multiple tunnels on your CPE, each to a corresponding headend. In the table below, "User" is you/your company.

Parameter	Source	Example Value
`${ipAddress1}`	Console/API	129.146.12.52
`${sharedSecret1}`	Console/API	(long string)

Parameter	Source	Example Value
`${ipAddress2}`	Console/API	129.146.13.52
`${sharedSecret2}`	Console/API	(long string)
`${cpePublicIpAddress}`	User	1.2.3.4
`${VcnCidrBlock}`	User	10.0.0.0/20

ISAKMP Policy Options

Parameter	Recommended Value
ISAKMP protocol version	Version 1
Exchange type	Main mode
Authentication method	Pre-shared keys
Encryption	AES-256-cbc
Authentication algorithm	SHA-256
Diffie-Hellman Group	Group 5
IKE session key lifetime	28800 seconds (8 hours)

IPSec Policy Options

Parameter	Recommended Value
IPSec protocol	ESP, tunnel-mode
Encryption	AES-256-cbc
Authentication algorithm	HMAC-SHA1-96

Parameter	Recommended Value
Diffie-Hellman Group	Group 5
Perfect Forward Secrecy	Enabled
IPSec session key lifetime	3600 seconds (1 hour)

Security Parameter Index

The values for the Security Parameter Index (SPI) depend on whether your CPE supports route-based tunnels or policy-based tunnels. For more information about the correct SPI values to use, see Route-Based Versus Policy-Based IPSec.

CPE Configuration

ISAKMP and IPSec Configuration

```
tunnel select 1
description tunnel OCI-VPN1
ipsec tunnel 1
  ipsec sa policy 1 1 esp aes256-cbc sha-hmac
  ipsec ike duration ipsec-sa 1 3600
  ipsec ike duration isakmp-sa 1 28800
  ipsec ike encryption 1 aes256-cbc
  ipsec ike group 1 modp1536
  ipsec ike hash 1 sha256
  ipsec ike keepalive log 1 off
  ipsec ike keepalive use 1 on dpd 5 4
  ipsec ike local address 1 ${cpePublicIpAddress}
  ipsec ike local id 1 0.0.0.0/0
  ipsec ike nat-traversal 1 on
  ipsec ike pfs 1 on
  ipsec ike pre-shared-key 1 text ${sharedSecret1}
  ipsec ike remote address 1 ${ipAddress1}
  ipsec ike remote id 1 0.0.0.0/0
ip tunnel tcp mss limit auto
tunnel enable 1

tunnel select 2
```

```
description tunnel OCI-VPN2
ipsec tunnel 2
  ipsec sa policy 2 2 esp aes256-cbc sha-hmac
  ipsec ike duration ipsec-sa 2 3600
  ipsec ike duration isakmp-sa 2 28800
  ipsec ike encryption 2 aes256-cbc
  ipsec ike group 2 modp1536
  ipsec ike hash 2 sha256
  ipsec ike keepalive log 2 off
  ipsec ike keepalive use 2 on dpd 5 4
  ipsec ike local address 2 ${cpePublicIpAddress}
  ipsec ike local id 2 0.0.0.0/0
  ipsec ike nat-traversal 2 on
  ipsec ike pfs 2 on
  ipsec ike pre-shared-key 2 text ${sharedSecret2}
  ipsec ike remote address 2 ${ipAddress2}
  ipsec ike remote id 2 0.0.0.0/0
ip tunnel tcp mss limit auto
tunnel enable 2

ipsec auto refresh on
```

STATIC ROUTES CONFIGURATION

```
ip route ${VcnCidrBlock} gateway tunnel 1 hide gateway tunnel 2 hide
```

Verified CPE Devices

The following devices or software have been verified for use with an Oracle Cloud Infrastructure IPSec VPN.

If your CPE device is not on this list, see the generic CPE configuration information.

Vendor	Device	Minimum Software Version	Configuration
Check Point	2200	R75.40	Check Point
Cisco	ASA 5500	9.7.1 (recommended)	Cisco ASA Configuration Options
Cisco	2921	IOS Version 15.4(3)M3	Cisco IOS
FortiGate	FortiGate-VM	v5.4.1,build1064 (GA)	FortiGate
Juniper	MX 240	Junos 15.1	Juniper MX
Juniper	SRX 240	12.1X44-D35.5	Juniper SRX
Libreswan (or Openswan)		3.23-4	Libreswan
Palo Alto	PA-500	PanOS version 6.0.6	Palo Alto
WatchGuard	Firebox	Fireware v12	WatchGuard
Yamaha	RTX1210	Firmware Rev.14.01.28	Yamaha RTX Series
Yamaha	RTX830	Firmware Rev.15.02.03	Yamaha RTX Series

Working with VPN Connect

This topic contains some details about working with VPN Connect and the related components. Also see these topics:

- VPN Connect Overview
- Setting Up VPN Connect

- Configuring Your CPE
- VPN Connect FAQ



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Viewing Tunnel Status and Configuration

When you successfully create the IPSec connection, Oracle produces important configuration information for each of the resulting IPSec tunnels. For example, see task 2h in the overall setup process. You can view that information and the status of the tunnels at any time. This includes the BGP status if the tunnel is configured to use BGP dynamic routing.

To view the status and configuration information for the IPSec tunnels

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **IPSec Connections**.
   A list of the IPSec connections in the compartment that you're viewing is displayed. If you don't see the one you're looking for, verify that you're viewing the correct compartment (select from the list on the left side of the page).
2. Click the IPSec connection you're interested in.
   Each tunnel's details are displayed, including the IPSec status, the BGP status (if the tunnel uses BGP dynamic routing), and the Oracle VPN IP address (the VPN headend).
3. To view a tunnel's shared secret:
   a. Click the tunnel you're interested in.
   b. Next to the **Shared Secret** field, click **Show**.

Changing the Static Routes

You can change the static routes for an existing IPSec connection. You can provide up to 10 static routes.

Remember that an IPSec connection can use either static routing or BGP dynamic routing. You associate the static routes with the overall IPSec connection and not the individual tunnels. If an IPSec connection has static routes associated with it, Oracle uses them for routing a tunnel's traffic *only* if the tunnel itself is configured to use static routing. If it's configured to use BGP dynamic routing, the IPSec connection's static routes are ignored.



Important

The IPSec connection goes down while it is reprovisioned with your static route changes.

To edit the static routes

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **IPSec Connections**.
   A list of the IPSec connections in the compartment that you're viewing is displayed. If you don't see the one you're looking for, verify that you're viewing the correct compartment (select from the list on the left side of the page).
2. For the IPSec connection you're interested in, click the Actions icon (three dots), and then click **Edit**.
   The current static routes are displayed.
3. Make your changes and click **Save Changes**.

Changing the CPE IKE Identifier That Oracle Uses

If your CPE is behind a NAT device, you might need to give Oracle your CPE IKE identifier. You can either specify it when you create the IPSec connection, or later edit the IPSec connection

and change the value. Oracle expects the value to be an IP address or fully qualified domain name (FQDN). When you specify the value, you also specify which type it is.



Important

The IPSec connection goes down while it is reprovisioned to use your CPE IKE identifier.

To change the CPE IKE identifier that Oracle uses

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **IPSec Connections**.
   A list of the IPSec connections in the compartment that you're viewing is displayed. If you don't see the one you're looking for, verify that you're viewing the correct compartment (select from the list on the left side of the page).
2. For the IPSec connection you're interested in, click the Actions icon (three dots), and then click **Edit**.
   The current CPE IKE identifier that Oracle is using is displayed at the bottom of the dialog.
3. Enter your new values for **CPE IKE Identifier Type** and **CPE IKE Identifier**, and then click **Save Changes**.

Changing the Shared Secret That an IPSec Tunnel Uses

When you set up an IPSec VPN, by default Oracle provides each tunnel's shared secret (also called the pre-shared key). You might have a particular shared secret that you want to use instead. You can specify each tunnel's shared secret when you create the IPSec connection, or you can edit the tunnels and provide each new shared secret then. For the shared secret, only numbers, letters, and spaces are allowed. Oracle recommends using a different shared secret for each tunnel.



Important

When you change a tunnel's shared secret, both the overall IPSec connection and the tunnel go into the Provisioning state while the tunnel is reprovisioned with the new shared secret. The other tunnel in the IPSec connection remains in the Available state. However, while the first tunnel is being reprovisioned, you cannot change the second tunnel's configuration.

To change the shared secret that an IPSec tunnel uses

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **IPSec Connections**.
   A list of the IPSec connections in the compartment that you're viewing is displayed. If you don’t see the one you're looking for, verify that you’re viewing the correct compartment (select from the list on the left side of the page).
2. Click the IPSec connection you're interested in.
3. Click the tunnel you're interested in.
4. Next to the **Shared Secret** field, click **Edit**.
5. Enter your new value. Only numbers, letters, and spaces are allowed.
6. Click **Save Changes**.

Changing from Static Routing to BGP Dynamic Routing

If you want to change an existing IPSec VPN from using static routing to using BGP dynamic routing, follow the process in this section.



Warning

When you change a tunnel's routing type, the tunnel's IPSec status does not change during reprovisioning. However, routing through the tunnel is affected. Traffic is temporarily disrupted until your network engineer configures your CPE device in accordance with the routing type change. **If your existing IPSec VPN is currently configured to use only a single tunnel, this process will disrupt your connection to Oracle.** If your IPSec VPN instead uses multiple tunnels, Oracle recommends reconfiguring one tunnel at a time to avoid disrupting your connection to Oracle.

To change from static routing to BGP dynamic routing

Prerequisites:

- You've read this section: Routing for the Oracle IPSec VPN
- You've gathered the necessary BGP routing information:
  - Your network's ASN
  - For each tunnel: The BGP IP address for each end of the tunnel (the two addresses for a given tunnel must be a pair from a /30 or /31 subnet, and they must be part of the IPSec VPN's encryption domain)

Repeat the following process for each tunnel in the IPSec connection:

1. Reconfigure the tunnel's routing type from static routing to BGP dynamic routing:
   a. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **IPSec Connections**.
   b. Click the IPSec connection you're interested in.

The tunnels are listed, and the status for each tunnel is shown. The **BGP Status** for the tunnel you're interested in should show only a hyphen (no value), which means that the tunnel is currently configured to use static routing.

c. Click the tunnel to view all of its details.

d. Click **Edit**.

e. Do the following:

- **Routing Type:** Select the radio button for **BGP Dynamic Routing**.
- **BGP ASN:** Enter your network's BGP ASN.
- **Inside Tunnel Interface - CPE:** Enter the BGP IP address with subnet mask (either /30 or /31) for the CPE end of the tunnel. For example: 10.0.0.16/31.
- **Inside Tunnel Interface - Oracle:** Enter the BGP IP address with subnet mask (either /30 or /31) for the Oracle end of the tunnel. For example: 10.0.0.17/31.

f. Click **Save Changes**.

The tunnel's **BGP Status** changes to Down.

2. Have your network engineer update your CPE device's tunnel configuration to use BGP.
3. On your side of the connection, confirm that the tunnel's BGP session is in an established state. If it is not, make sure you've configured the correct IP addresses for the tunnel in the Oracle Console and also for your CPE device.
4. In the Oracle Console, confirm that the tunnel's **BGP Status** is now Up.
5. Confirm that your CPE device is learning routes from Oracle, and your CPE device is advertising routes to Oracle. If you want to re-advertise the Oracle routes from BGP back to your on-premises network, make sure your CPE device is configured accordingly. Your existing policy to advertise the static routes to your on-premises network may not work for the BGP learned routes.
6. Ping the Oracle BGP IP address from your side of the connection to confirm that traffic is flowing.

After you've confirmed the first tunnel is up and running with BGP, repeat the process for the second tunnel.



Important

As noted in Routing for the Oracle IPSec VPN, the static routes that are still configured for the overall IPSec connection do NOT override the BGP routing. Those static routes are ignored when Oracle routes traffic through a tunnel that is configured to use BGP.

Also, you can change a tunnel's routing type back to static routing if necessary. You might do this if the scheduled downtime window for the CPE device is ending soon and you're having trouble establishing the BGP session. When you switch back to static routing, make sure the overall IPSec connection still has your desired static routes configured.

Disabling or Terminating the IPSec VPN

If you want to disable the IPSec VPN between your on-premises network and VCN, you can simply detach the DRG from the VCN instead of deleting the IPSec connection. If you're also using the DRG with FastConnect, detaching the DRG would also interrupt the flow of traffic over FastConnect.

You can delete the IPSec connection. However, if you later want to re-establish it, your network engineer would have to configure your CPE device again with a new set of tunnel configuration information from Oracle.

If you want to permanently delete the entire IPSec VPN, you must first terminate the IPSec connection. Then you can delete the CPE object. If you're not using the DRG for another connection to your on-premises network, you can detach it from the VCN and then delete it.

To delete an IPSec connection

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **IPSec Connections**.
   A list of the IPSec connections in the compartment you're viewing is displayed. If you don't see the one you're looking for, verify that you're viewing the correct compartment (select from the list on the left side of the page).
2. Click the IPSec connection you're interested in.
3. Click **Terminate**.
4. Confirm the deletion when prompted.

The IPSec connection will be in the Terminating state for a short period while it's being deleted.

To delete a CPE object

Prerequisite: There must not be an IPSec connection between the CPE object and a DRG.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Customer-Premises Equipment**.
   A list of the CPE objects in the compartment you're viewing is displayed. If you don't see the one you're looking for, verify that you're viewing the correct compartment (select from the list on the left side of the page).
2. For the CPE object that you want to delete, click the Actions icon (three dots), and then click **Delete**.
3. Confirm the deletion when prompted.

The object will be in the Terminating state for a short period while it's being deleted.

Managing Tags for an IPSec Connection or CPE Object

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource

later with the desired tags. For general information about applying tags, see Resource Tags.

To manage tags for an IPSec connection

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **IPSec Connections**.
   A list of the IPSec connections in the compartment you're viewing is displayed. If you don't see the one you're looking for, verify that you're viewing the correct compartment (select from the list on the left side of the page).
2. Click the IPSec connection you're interested in.
3. Click the **Tags** tab to view or edit the existing tags. Or click **Add tags** to add new ones.

To manage tags for a CPE object

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Customer-Premises Equipment**.
   A list of the CPE objects in the compartment you're viewing is displayed. If you don't see the one you're looking for, verify that you're viewing the correct compartment (select from the list on the left side of the page).
2. Click the CPE object you're interested in.
3. Click the **Tags** tab to view or edit the existing tags. Or click **Apply tag(s)** to add new ones.

Managing Your DRG

For tasks related to DRGs, see Dynamic Routing Gateways (DRGs).

VPN Connect FAQ

Can I customize configuration parameters such as IKE ID or IPSec VPN tunnel lifetime on the DRG?

No. Oracle has predetermined the configuration parameters that work with the IPSec VPN service. Your IPsec VPN can't be established if there is a mismatch.

If your CPE is behind a NAT device, you can provide Oracle with your CPE's IKE identifier so that Oracle can use the same value on the Oracle side. For more information, see Changing the CPE IKE Identifier That Oracle Uses.

Can Oracle initiate the IPsec VPN connection?

No. You must initiate it from your end.

Is an IPSec VPN supported over FastConnect?

No, FastConnect and the IPSec VPN are two different services.

Can packets from the VCN be sourced with the public IP of the DRG? In other words, can the DRG source the NAT VCN traffic?

No. Packets from the VCN have the private IP address of the instance as a source IP address. Oracle cannot change the source IP address to the private or public IP address of the DRG.

Oracle has provided two VPN endpoints to build tunnels to. Does Oracle route the same network over both tunnels?

Yes. Traffic for all the subnets in the VCN attached to your DRG is routed over both tunnels.

How many VPN tunnels can I have from a single CPE device?

You can have a maximum of eight tunnels from a unique CPE IP address per region. If you want more than eight tunnels, either use a different IP address for the additional ones, or use a different CPE device (recommended).

I'm having trouble with my IPSec connection. What can I do?

See VPN Connect Troubleshooting.

Using the API for VPN Connect

This topic lists the Networking service API operations for managing VPN Connect components.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

To manage your VCN and subnets, use these operations:

- ListVcns
- CreateVcn
- GetVcn
- UpdateVcn
- DeleteVcn

- ListSubnets
- CreateSubnet
- GetSubnet
- UpdateSubnet
- DeleteSubnet

To manage your DRG, use these operations:

- ListDrgs
- CreateDrg
- GetDrg
- UpdateDrg
- DeleteDrg
- ListDrgAttachments
- CreateDrgAttachment: This operation attaches a DRG to a VCN and results in a `DrgAttachment` object with its own OCID.
- GetDrgAttachment
- UpdateDrgAttachment
- DeleteDrgAttachment: This operation detaches a DRG from a VCN by deleting the `DrgAttachment` object.

To manage routing for your VCN, use these operations:

- ListRouteTables
- GetRouteTable
- UpdateRouteTable
- CreateRouteTable
- DeleteRouteTable

To manage security lists for your VCN, use these operations:

- ListSecurityLists
- GetSecurityList
- UpdateSecurityList
- CreateSecurityList
- DeleteSecurityList

To manage your CPEs, use these operations:

- ListCpes
- CreateCpe
- GetCpe
- UpdateCpe
- DeleteCpe

To manage your IPSec connections, use these operations:

- ListIPSecConnections
- CreateIPSecConnection: Use this operation to get the configuration information for each tunnel, including the IP address of the DRG (the VPN headend) and the shared secret. See Configuring Your CPE.
- GetIPSecConnection
- UpdateIPSecConnection
- DeleteIPSecConnection
- GetIPSecConnectionDeviceStatus: Use this operation to determine the status of the IPSec tunnels (up or down).
- GetIPSecConnectionDeviceConfig: Use this operation to get the configuration information for each tunnel.

VPN Connect Troubleshooting

This topic covers troubleshooting techniques for an IPSec VPN that has issues.

Some of the troubleshooting techniques assume that you are a network engineer with access to your CPE device's configuration.

General Issues

IPSec tunnel is DOWN

Check these items:

- **Basic configuration:** The IPSec tunnel consists of both phase-1 (ISAKMP) and phase-2 (IPSec) configuration. Confirm that both are configured correctly on your CPE device. See the configuration appropriate for your CPE device:

List of configurations

- Generic CPE Configuration Information
- Check Point
- Cisco ASA:
  - Cisco ASA Configuration Options
  - Cisco ASA: Route-Based
  - Cisco ASA: Policy-Based Without VPN Filters
  - Cisco ASA: Policy-Based with VPN Filters
- Cisco IOS
- FortiGate
- Juniper MX
- Juniper SRX
- Libreswan
- Openswan
- Palo Alto

- WatchGuard
- Yamaha RTX Series

- **Local and remote proxies:** If you're using a policy-based configuration, check if your CPE is configured with more than one pair of local and remote proxy IDs (subnets). The Oracle VPN router supports only one pair. If your CPE has more than one pair, update the configuration to include only one pair, and choose one of the following two options:

Option	Local Proxy ID	Remote Proxy ID
1	ANY (or 0.0.0.0/0)	ANY (or 0.0.0.0/0)
2	On-premises CIDR (an aggregate that covers all the subnets of interest)	VCN's CIDR

- **NAT device:** If the CPE is behind a NAT device, the CPE IKE identifier configured on your CPE might not match the CPE IKE identifier Oracle is using (the public IP address of your CPE). If your CPE does not support setting the CPE IKE identifier on your end, please file a ticket at My Oracle Support so Oracle can update the CPE IKE identifier on the Oracle end to match yours.
- **NAT-T:** The Oracle VPN router currently does not support NAT-T. If your CPE is currently configured for NAT-T, disable NAT-T and then re-initiate the IPSec connection on your CPE.

IPSec tunnel is UP, but no traffic is passing through

Check these items:

- **Phase 2 (IPSec) configuration:** Confirm that the phase 2 (IPSec) parameters are configured correctly on your CPE device. See the configuration appropriate for your CPE device:

List of configurations

- Generic CPE Configuration Information
- Check Point
- Cisco ASA:
  - Cisco ASA Configuration Options
  - Cisco ASA: Route-Based
  - Cisco ASA: Policy-Based Without VPN Filters
  - Cisco ASA: Policy-Based with VPN Filters
- Cisco IOS
- FortiGate
- Juniper MX
- Juniper SRX
- Libreswan
- Openswan
- Palo Alto
- WatchGuard
- Yamaha RTX Series

- **NAT-T:** The Oracle VPN router currently does not support NAT-T. If your CPE is currently configured for NAT-T, disable NAT-T and then re-initiate the IPSec connection on your CPE.
- **VCN security lists:** Ensure you've set up the VCN security lists to allow the desired traffic (both ingress and egress rules). Note that the VCN's default security list does not allow ping traffic (ICMP type 8 and ICMP type 0). You must add the appropriate ingress and egress rules to allow ping traffic.
- **Firewall rules:** Ensure that your firewall rules allow both ingress and egress traffic with the Oracle VPN headend IPs and the VCN CIDR block.

- **Asymmetric routing:** Oracle uses asymmetric routing across the multiple tunnels that make up the IPSec VPN connection. Even if you configure one tunnel as primary and another as backup, traffic from your VCN to your on-premises network can use any tunnel that is "up" on your device. Configure your firewalls accordingly. Otherwise, ping tests or application traffic across the connection will not reliably work.
- **Cisco ASA:** Do not use the `originate-only` option with an Oracle IPSec VPN tunnel. It causes the tunnel's traffic to be inconsistently blackholed. The command is only for tunnels between two Cisco devices. Here's an example of the command that you should NOT use for the Oracle IPSec VPN tunnels: `crypto map <map name> <sequence number> set connection-type originate-only`

IPSec tunnel is UP, but traffic is passing in only one direction

Check these items:

- **Asymmetric routing:** Oracle uses asymmetric routing across the multiple tunnels that make up the IPSec VPN connection. Even if you configure one tunnel as primary and another as backup, traffic from your VCN to your on-premises network can use any tunnel that is "up" on your device. Configure your firewalls accordingly. Otherwise, ping tests or application traffic across the connection will not reliably work.
- **Single tunnel preferred:** If you want to use only one of the tunnels, make sure that you have the proper policy or routing in place on the CPE to prefer that tunnel.
- **Multiple IPSec connections:** If you have multiple IPSec connections with Oracle, make sure to specify more specific static routes for the preferred IPSec connection.
- **VCN security lists:** Ensure that your VCN security lists allow traffic in both directions (ingress and egress).
- **Firewall rules:** Ensure that your firewall rules allow traffic in *both* directions with the Oracle VPN headend IPs and the VCN CIDR block.

For an IPSec VPN with a Policy-Based Configuration

IPSec tunnel is DOWN

Check these items:

- **Basic configuration:** The IPSec tunnel consists of both phase-1 (ISAKMP) and phase-2 (IPSec) configuration. Confirm that both are configured correctly on your CPE device. See the configuration appropriate for your CPE device:

List of configurations

- Generic CPE Configuration Information
- Check Point
- Cisco ASA:
  - Cisco ASA Configuration Options
  - Cisco ASA: Route-Based
  - Cisco ASA: Policy-Based Without VPN Filters
  - Cisco ASA: Policy-Based with VPN Filters
- Cisco IOS
- FortiGate
- Juniper MX
- Juniper SRX
- Libreswan
- Openswan
- Palo Alto
- WatchGuard
- Yamaha RTX Series

- **Local and remote proxies:** If you're using a policy-based configuration, check if your CPE is configured with more than one pair of local and remote proxy IDs (subnets). The Oracle VPN router supports only one pair. If your CPE has more than one pair, update the configuration to include only one pair, and choose one of the following two options:

Option	Local Proxy ID	Remote Proxy ID
1	ANY (or 0.0.0.0/0)	ANY (or 0.0.0.0/0)
2	On-premises CIDR (an aggregate that covers all the subnets of interest)	VCN's CIDR

- **NAT device:** If the CPE is behind a NAT device, the CPE IKE identifier configured on your CPE might not match the CPE IKE identifier Oracle is using (the public IP address of your CPE). If your CPE does not support setting the CPE IKE identifier on your end, please file a ticket at My Oracle Support so Oracle can update the CPE IKE identifier on the Oracle end to match yours.
- **Cisco ASA:** Do not use the `originate-only` option with an Oracle IPSec VPN tunnel. It causes the tunnel's traffic to be inconsistently blackholed. The command is only for tunnels between two Cisco devices. Here's an example of the command that you should NOT use for the Oracle IPSec VPN tunnels: `crypto map <map name> <sequence number> set connection-type originate-only`

IPSec tunnel is UP but keeps flapping

Check these items:

- **Initiation of connection:** Ensure that your CPE device is initiating the connection.
- **Local and remote proxies:** If you're using a policy-based configuration, check if your CPE is configured with more than one pair of local and remote proxy IDs (subnets). The Oracle VPN router supports only one pair. If your CPE has more than one pair, update

the configuration to include only one pair, and choose one of the following two options:

Option	Local Proxy ID	Remote Proxy ID
1	ANY (or 0.0.0.0/0)	ANY (or 0.0.0.0/0)
2	On-premises CIDR (an aggregate that covers all the subnets of interest)	VCN's CIDR

- **Interesting traffic at all times:** In general, Oracle recommends having interesting traffic running through the IPSec tunnels at all times if your CPE supports it.
  Certain Cisco ASA versions require the SLA monitor to be configured, which keeps interesting traffic running through the IPSec tunnels. See:
  - For Cisco ASA: Policy-Based with VPN Filters: SLA Monitoring Configuration
  - For Cisco ASA: Policy-Based Without VPN Filters: SLA Monitoring Configuration

IPSec tunnel is UP but traffic is unsteady

Check these items:

- **Local and remote proxies:** If you're using a policy-based configuration, check if your CPE is configured with more than one pair of local and remote proxy IDs (subnets). The Oracle VPN router supports only one pair. If your CPE has more than one pair, update the configuration to include only one pair, and choose one of the following two options:

Option	Local Proxy ID	Remote Proxy ID
1	ANY (or 0.0.0.0/0)	ANY (or 0.0.0.0/0)
2	On-premises CIDR (an aggregate that covers all the subnets of interest)	VCN's CIDR

- **Interesting traffic at all times:** In general, Oracle recommends having interesting traffic running through the IPSec tunnels at all times if your CPE supports it.
  Certain Cisco ASA versions require the SLA monitor to be configured, which keeps interesting traffic running through the IPSec tunnels. See:
  - For Cisco ASA: Policy-Based with VPN Filters: SLA Monitoring Configuration
  - For Cisco ASA: Policy-Based Without VPN Filters: SLA Monitoring Configuration

BGP Session Troubleshooting

BGP status is DOWN

Check these items:

- **IPSec status:** For the BGP session to be up, the IPSec tunnel itself must be up.
- **BGP address:** Verify that both ends of the tunnel are configured with the correct BGP peering IP address.
- **ASN:** Verify that both ends of the tunnel are configured with the correct BGP local ASN and Oracle ASN (31898).
- **MD5:** Verify that MD5 authentication is disabled or not configured on your CPE device. The Oracle IPSec VPN does not support MD5 authentication.
- **Firewalls:** Verify that your on-premises firewall or access control lists are not blocking the following ports:

- TCP port 179 (BGP)
- UDP port 500 (IKE)
- IP protocol port 50 (ESP)

If your CPE device's firewall is blocking TCP port 179 (BGP), the BGP neighborship will be down. Traffic cannot flow through the tunnel because the CPE device and Oracle router do not have any routes.

BGP status is flapping

Check these items:

- **IPSec status:** For the BGP session to be up and not flapping, the IPSec tunnel itself must be up and not flapping.
- **Maximum prefixes:** Verify that you are advertising no more than 2000 prefixes. If you're advertising more, BGP won't be established.

BGP status is UP, but no traffic is passing through

Check these items:

- **VCN security lists:**Ensure you've set up the VCN security lists to allow the desired traffic (both ingress and egress rules). Note that the VCN's default security list does not allow ping traffic (ICMP type 8 and ICMP type 0). You must add the appropriate ingress and egress rules to allow ping traffic.
- **Correct routes on both ends:** Verify that you have received the correct VCN routes from Oracle and the CPE device is using those routes. Likewise, verify that you are advertising the correct on-premises network routes over the IPSec VPN, and the VCN route tables use those routes.

BGP status is UP, but traffic is passing in only one direction

Check these items:

- **VCN security lists:** Ensure that your VCN security lists allow traffic in both directions (ingress and egress).
- **Firewalls:** Verify that your on-premises firewall or access control lists are not blocking traffic to or from the Oracle end.
- **Asymmetric routing:** Oracle uses asymmetric routing. If you have multiple IPSec connections, make sure that your CPE device is configured for asymmetric route processing.
- **Redundant connections:** If you have redundant IPSec connections, make sure they're both advertising the same routes.

Redundant Connections

Keep in mind these important notes:

- FastConnect uses BGP dynamic routing. IPSec connections can use either static routing or BGP, or combination.
- For important details about routing and preferred routes when using redundant connections, see Routing Preferences When You Have Redundant Connections.
- You can use two IPSec connections for redundancy. If both IPSec connections have only a default route (0.0.0.0/0) configured, traffic will route to either of those connections because Oracle uses asymmetric routing. If you want one IPSec connection as primary and another one as backup, configure more-specific routes for the primary connection and less-specific routes (or the default route of 0.0.0.0/0) on the backup connection.

IPSec and FastConnect are both set up, but traffic is only passing through IPSec

Make sure to use more specific routes for the connection you want as primary. If you're using the same routes for both IPSec and FastConnect, see the discussion of routing preferences in Routing Preferences When You Have Redundant Connections.

Two on-premises data centers each have an IPSec connection to Oracle, but only one is passing traffic

Verify that both IPSec connections are up and ensure that you have asymmetric route processing enabled on the CPE.

If both IPSec connections have only a default route (0.0.0.0/0) configured, traffic will route to either of those connections because Oracle uses asymmetric routing. If you want one IPSec connection as primary and another one as backup, configure more-specific routes for the primary connection and less-specific routes (or the default route of 0.0.0.0/0) on the backup connection.

For more information about this type of setup, see Example Layout with Multiple Geographic Areas.

FastConnect

The following topics have information about setting up Oracle Cloud Infrastructure FastConnect between your on-premises network and virtual cloud network (VCN):

- FastConnect Overview
- FastConnect Requirements
- FastConnect: With an Oracle Provider
- FastConnect: With a Third-Party Provider
- FastConnect: Colocation with Oracle
- FastConnect Troubleshooting

FastConnect Overview

Oracle Cloud Infrastructure FastConnect provides an easy way to create a dedicated, private connection between your data center and Oracle Cloud Infrastructure. FastConnect provides higher-bandwidth options, and a more reliable and consistent networking experience compared to internet-based connections.

Uses for FastConnect

With FastConnect, you can choose to use *private peering*, *public peering*, or both.

- **Private peering:** To extend your existing infrastructure into a virtual cloud network (VCN) in Oracle Cloud Infrastructure (for example, to implement a hybrid cloud, or a lift and shift scenario). Communication across the connection is with IPv4 private addresses (typically RFC 1918).
- **Public peering:** To access public services in Oracle Cloud Infrastructure without using the internet. For example, Object Storage, the Oracle Cloud Infrastructure Console and APIs, or public load balancers in your VCN. Communication across the connection is with IPv4 public IP addresses. Without FastConnect, the traffic destined for public IP addresses would be routed over the internet. With FastConnect, that traffic goes over your private physical connection.

In general it's assumed you'll use private peering, and you *might* also use public peering. Most of this documentation is relevant to both, with specific details called out for private versus public.

How and Where to Connect

With FastConnect, there are different connectivity models to choose from.

Oracle Provider

- List of Oracle Cloud Infrastructure FastConnect providers
- Port speeds in 1-Gbps and 10-Gbps increments
- Instructions for integrating: FastConnect: With an Oracle Provider

Third-Party Provider

- Port speed of 10 Gbps per cross-connect
- Instructions for integrating: FastConnect: With a Third-Party Provider

Colocation with Oracle in an Oracle Cloud Infrastructure FastConnect Location

- List of Oracle Cloud InfrastructureFastConnect locations (see the FAQ for specific addresses)
- Port speed of 10 Gbps per cross-connect
- Instructions for integrating: FastConnect: Colocation with Oracle

The following table summarizes several important requirements for each connectivity model.

Requirement	With Oracle Provider	With Third-Party Provider	Colocation with Oracle
Routing requirements	Yes	Yes	Yes
BGP support	Yes	Yes	Yes
Layer 3 support	Recommended	Recommended	Recommended
Obtain a Letter of Authority (LOA) from Oracle	N/A	Yes	Yes
Network connectivity	Yes	Yes	N/A
Cross-connect	Yes (from the provider)	Yes	Yes
Redundant network connectivity	Recommended	Recommended	Recommended
Cloud connectivity solution architecture support	Recommended	Recommended	Recommended
FastConnect SKU	Yes	Yes	Yes

Requirement	With Oracle Provider	With Third-Party Provider	Colocation with Oracle
Oracle Cloud Infrastructure Console user login (IAM policy unique setup)	Yes	Yes	Yes
Tenancy established (see "Setting Up Your Tenancy" in the *Oracle Cloud Infrastructure Getting Started Guide*)	Yes	Yes	Yes

Concepts

Here are some important concepts to understand (also see the following diagrams):

FASTCONNECT

The general concept of a connection between your existing network and Oracle Cloud Infrastructure over a private physical network instead of the internet.

FASTCONNECT LOCATION

A specific Oracle data center where you can connect with Oracle Cloud Infrastructure.

METRO AREA

A geographical area (for example, Ashburn) with multiple FastConnect locations. All locations in a metro area connect to the same set of availability domains for resiliency in case of failure in a single location.

ORACLE PROVIDER

A network service provider that has integrated with Oracle in a FastConnect location. See the list of the Oracle providers in How and Where to Connect. If your provider is in the list, see FastConnect: With an Oracle Provider.

THIRD-PARTY PROVIDER

A network service provider that is NOT on the list of Oracle providers in How and Where to Connect. If you have a third-party provider and want to use FastConnect, see FastConnect: With a Third-Party Provider.

COLOCATION

The situation where your equipment is deployed into a FastConnect location. If your network service provider is not on the list of Oracle providers in How and Where to Connect, you must colocate.

CROSS-CONNECT

In a colocation or third-party provider scenario, this is the physical cable connecting your existing network to Oracle in the FastConnect location.

CROSS-CONNECT GROUP

In a colocation or third-party provider scenario, this is a link aggregation group (LAG) that contains at least one cross-connect. You can add additional cross-connects to a cross-connect group as your bandwidth needs increase. This is applicable only for colocation.

VIRTUAL CLOUD NETWORK (VCN)

Your virtual network in Oracle Cloud Infrastructure. You can use a VCN to extend your infrastructure into the cloud. For more information, see VCNs and Subnets.

DYNAMIC ROUTING GATEWAY (DRG)

A virtual edge router attached to your VCN. Necessary for private peering. The DRG is a single point of entry for private traffic coming in to your VCN, whether it's over FastConnect or an IPSec VPN. After creating the DRG, you must attach it to your VCN and add a route for the DRG in the VCN's route table to enable traffic flow. Instructions for everything are included in the sections that follow.

VIRTUAL CIRCUIT

An isolated network path that runs over one or more physical network connections to provide a single, logical connection between the edge of your existing network and Oracle

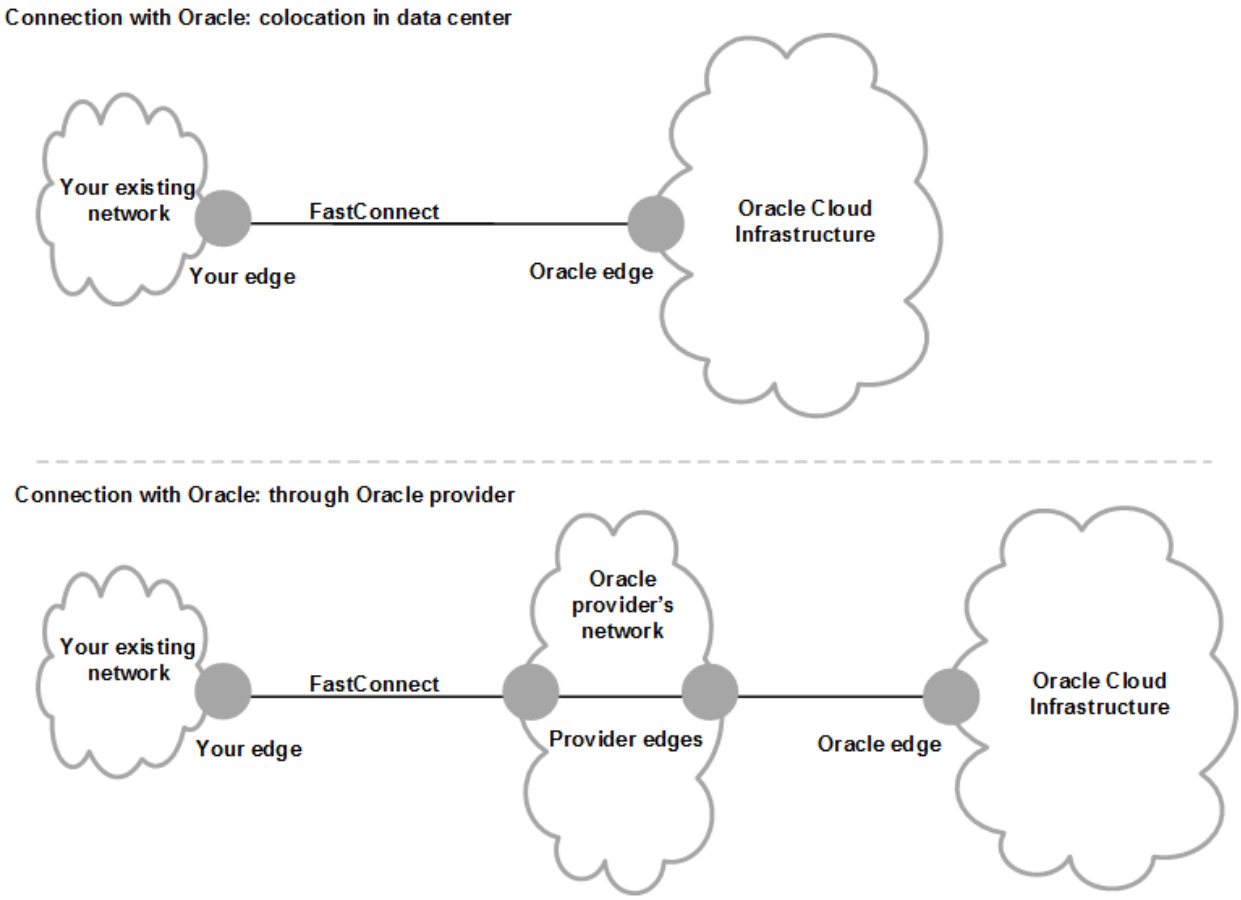
Cloud Infrastructure. *Private virtual circuits* support private peering, and *public virtual circuits* support public peering (see Uses for FastConnect). Each virtual circuit is made up of information shared between you and Oracle, as well as a provider (if you're connecting through an Oracle provider). You could have multiple private virtual circuits, for example, to isolate traffic from different parts of your organization (one virtual circuit for 10.0.1.0/24; another for 172.16.0.0/16), or to provide redundancy.

Basic Network Diagrams

The diagrams in this section introduce the basic logical and physical connections involved in FastConnect. Details specific to private versus public peering are called out.

General Concept of FastConnect

The following diagram illustrates the two ways to connect to Oracle with FastConnect: either through colocation with Oracle in the FastConnect location, or through an Oracle provider. In both cases, the connection goes between the edge of your existing network and Oracle.

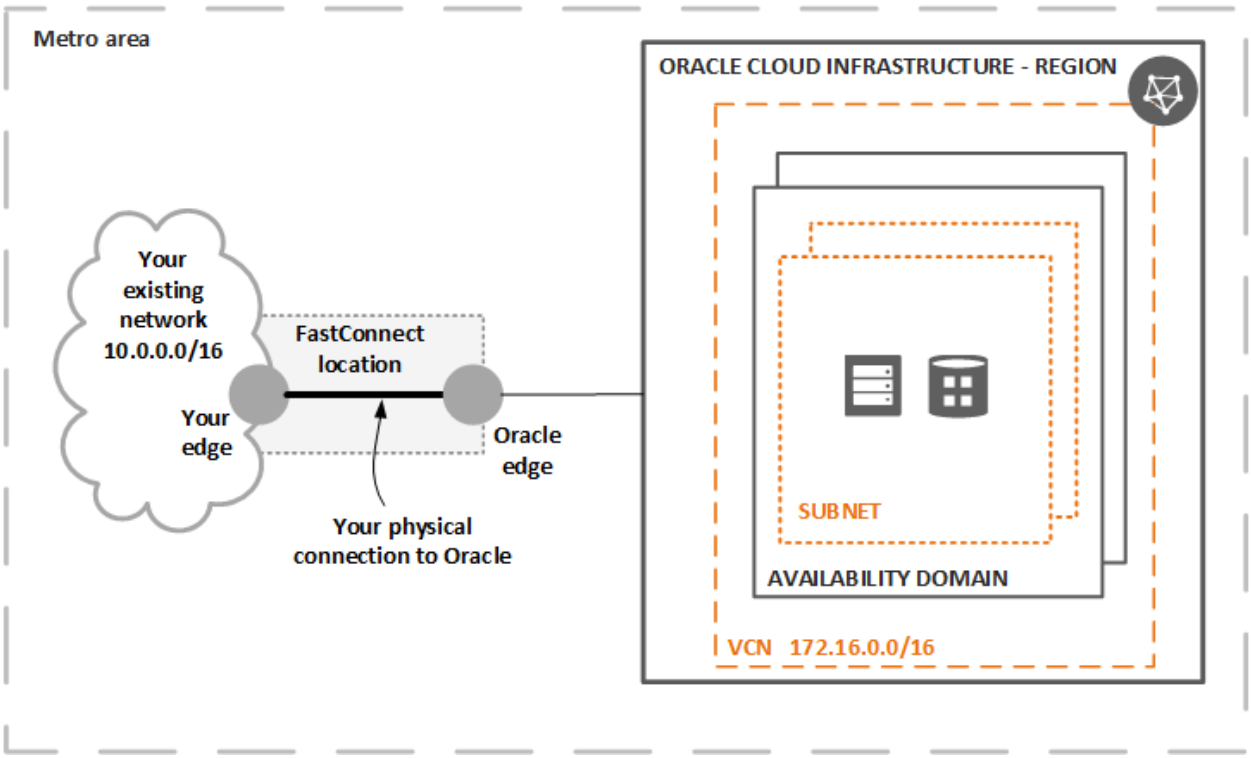


Physical Connection

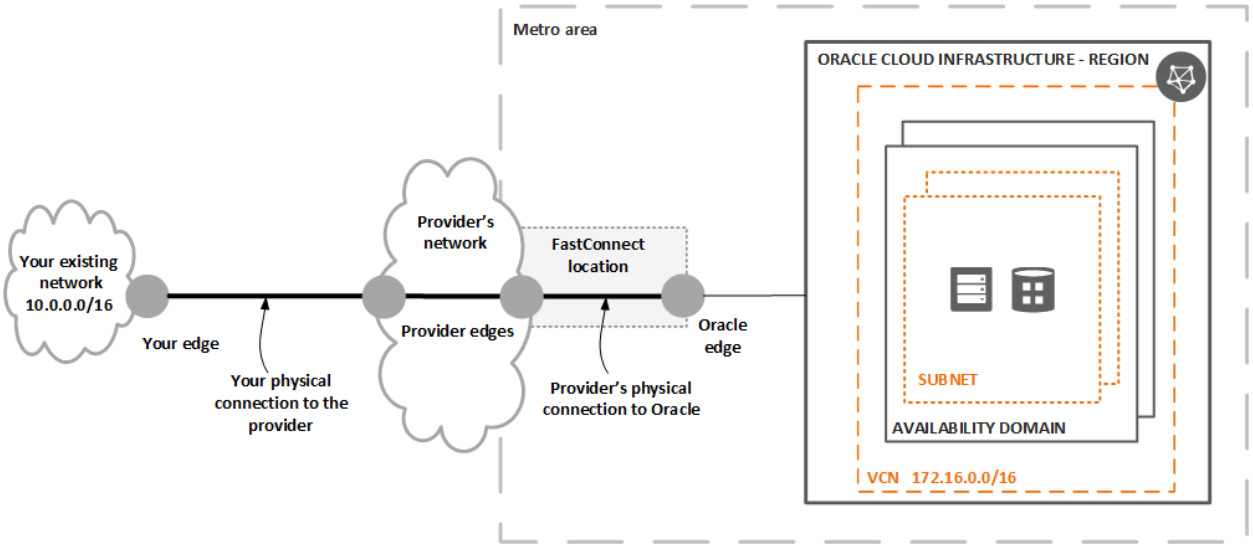
The next two diagrams give more detail about the physical connections. They also show the metro area that contains the FastConnect location, and a VCN within an Oracle Cloud

Infrastructure region.

The first diagram shows the colocation scenario, with your physical connection to Oracle within the FastConnect location.



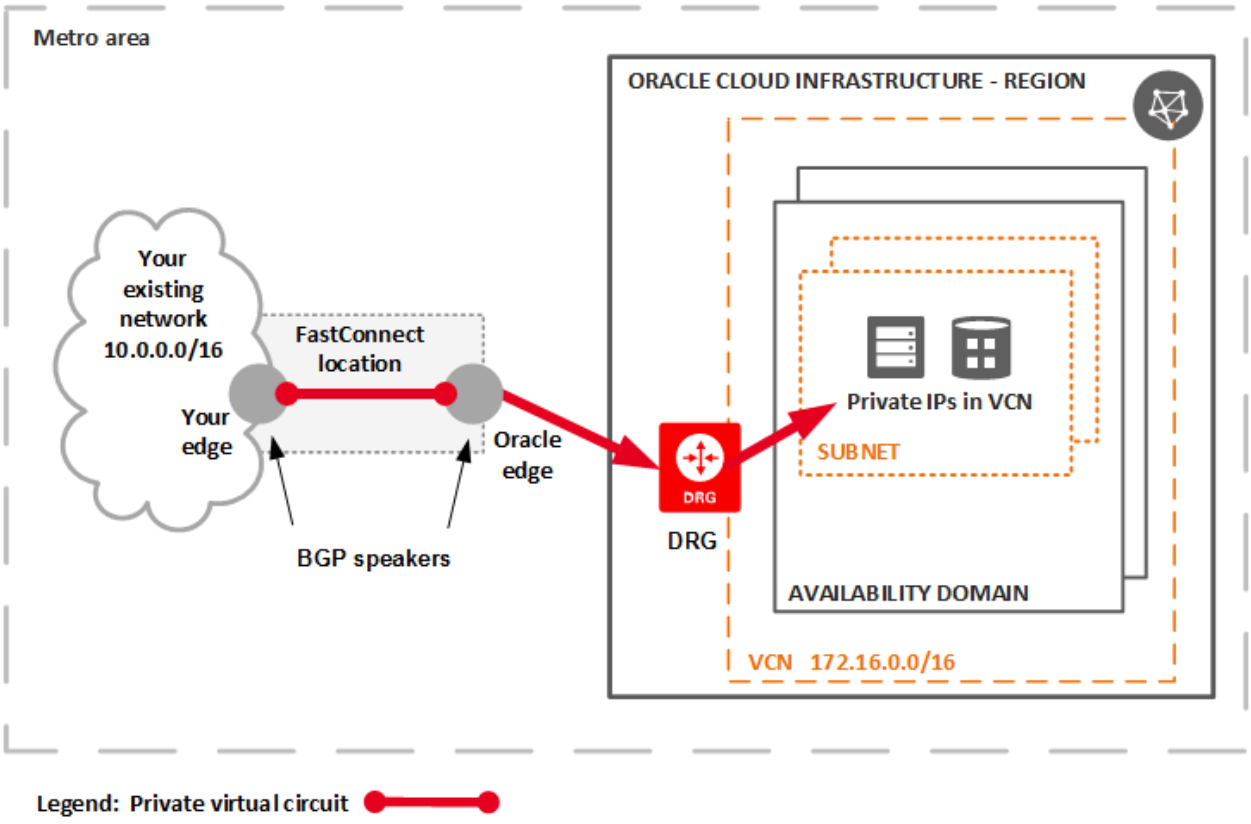
The next diagram shows a scenario with either an Oracle provider or third-party provider. It shows your physical connection to the provider, and the provider's physical connection to Oracle within the FastConnect location.



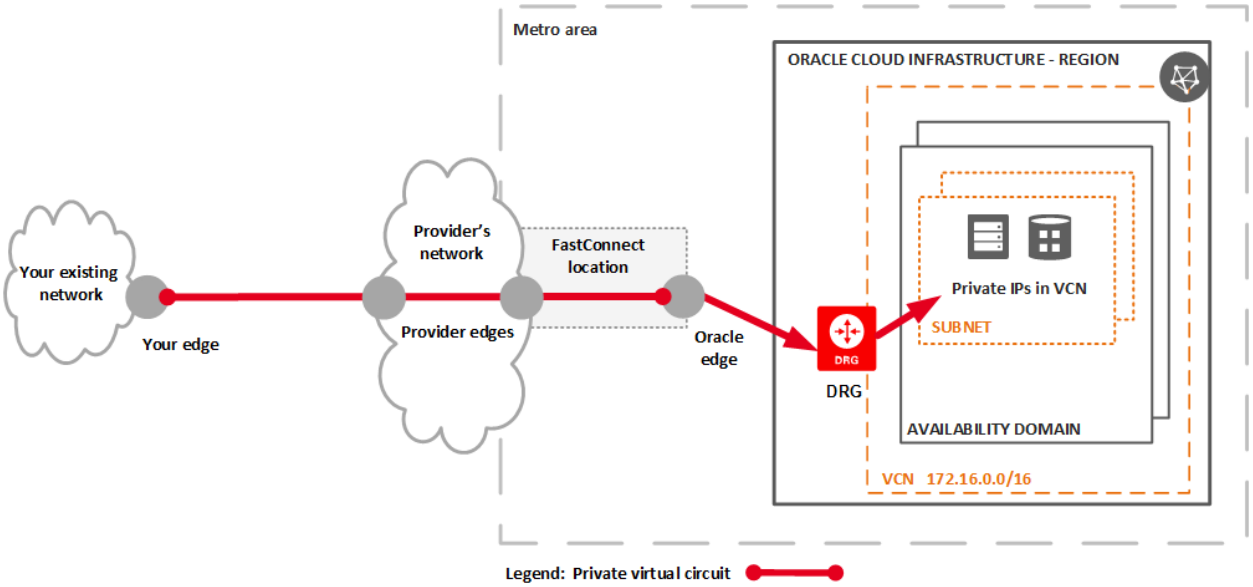
Logical Connection: Private Virtual Circuit

The next two diagrams show a private virtual circuit, which is a single, logical connection between your edge and Oracle Cloud Infrastructure by way of your DRG. Traffic is destined for private IP addresses in your VCN.

The first diagram shows the colocation scenario.



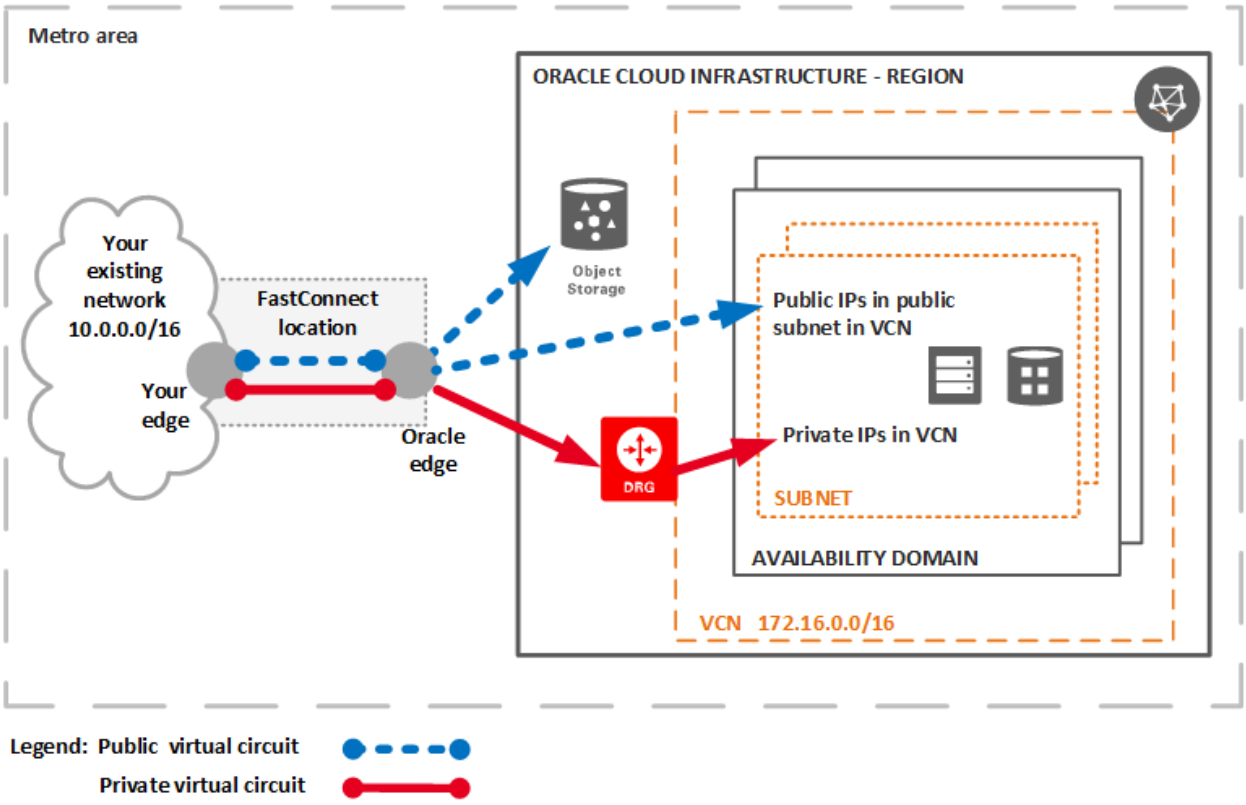
The next diagram shows the scenario with either an Oracle provider or third-party provider.



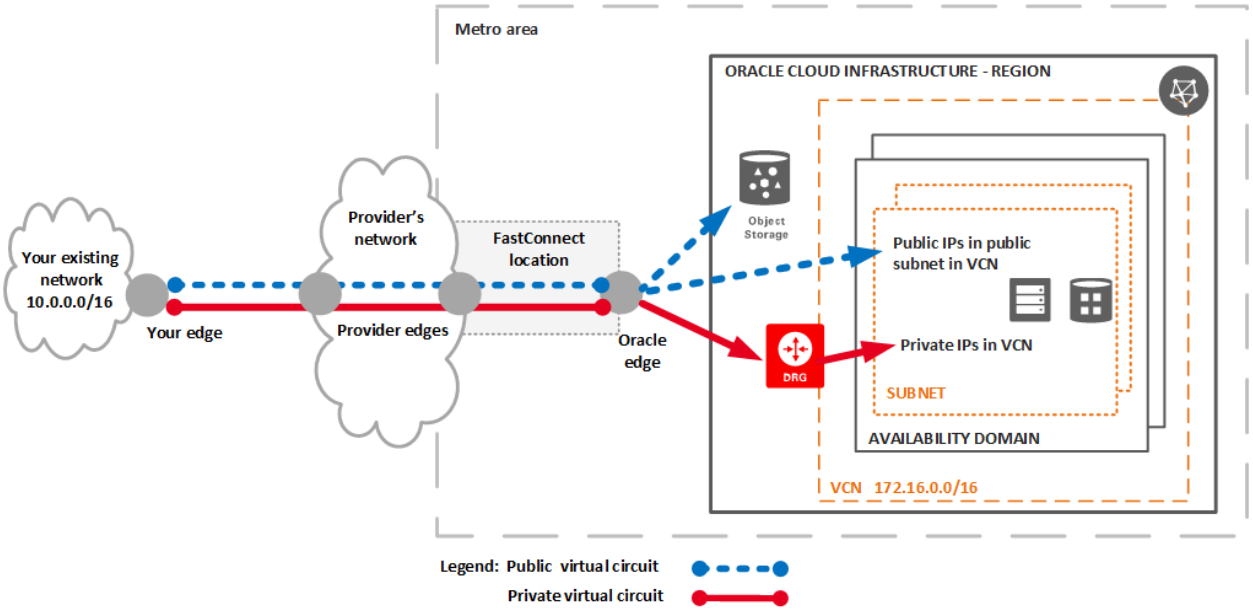
Logical Connection: Public Virtual Circuit

A public virtual circuit gives your existing network access to regional public IPv4 addresses in Oracle Cloud Infrastructure. For example, Object Storage, the Oracle Cloud Infrastructure Console and APIs, and public load balancers in your VCN. All communication across a public virtual circuit uses public IP addresses.

The first diagram shows the colocation scenario with both a private virtual circuit and a public virtual circuit. Notice that the DRG is not involved with the public virtual circuit, only the private virtual circuit.



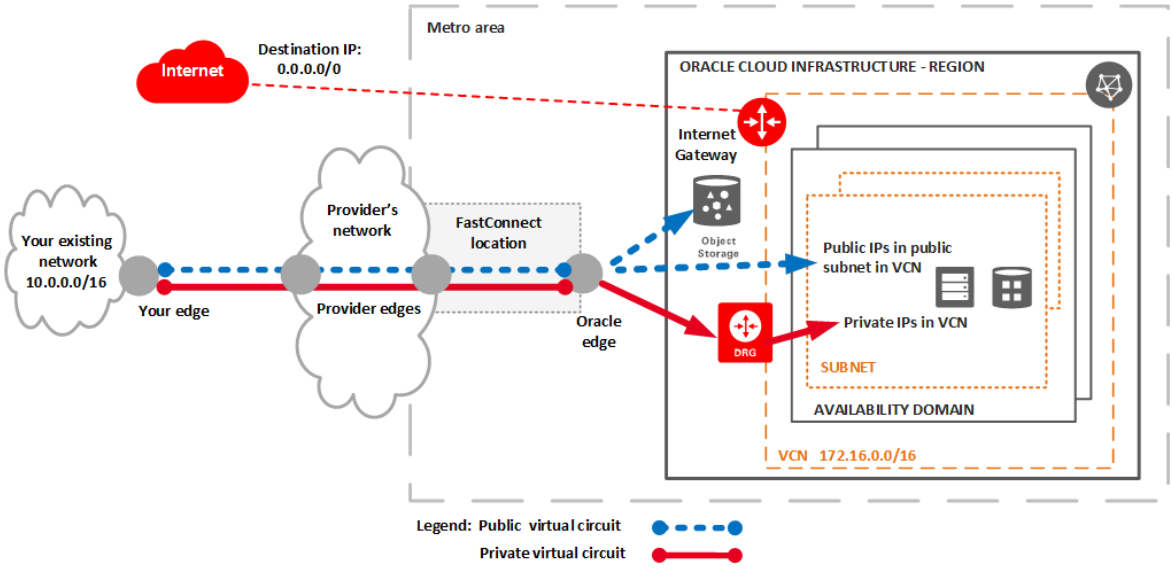
The next diagram shows the scenario with either an Oracle provider or third-party provider.



Here are a few basics to know about public virtual circuits:

- You choose which of your organization's public IP prefixes you want to use with the virtual circuit. Each prefix must be /31 or less specific. Oracle verifies your organization's ownership of each prefix before sending any traffic for it across the connection. Oracle's verification for a given prefix can take up to three business days. You can get the status of the verification of each prefix in the Oracle Console or API. **Oracle begins advertising the Oracle Cloud Infrastructure public IP addresses across the connection only after successfully verifying at least one of your public prefixes.**
- Your existing network will receive Oracle's public IP addresses through both FastConnect and your Internet Service Provider (ISP). When configuring your edge, make sure to give higher preference to FastConnect over your ISP, or you will not receive the benefits of FastConnect.

- You must configure your firewall rules to allow the desired traffic coming from the Oracle public IP addresses.
- Oracle prefers the most specific route when routing traffic from Oracle Cloud Infrastructure to other destinations. This means that when Oracle replies to traffic coming from one of your verified public prefixes, the reply travels over the FastConnect public virtual circuit, even if you have an internet gateway on your VCN. Replies to traffic from any public prefixes that are not on your list or not yet verified by Oracle still go through the internet gateway. For reference, this diagram shows the private and public virtual circuits as well as an internet gateway on the VCN:



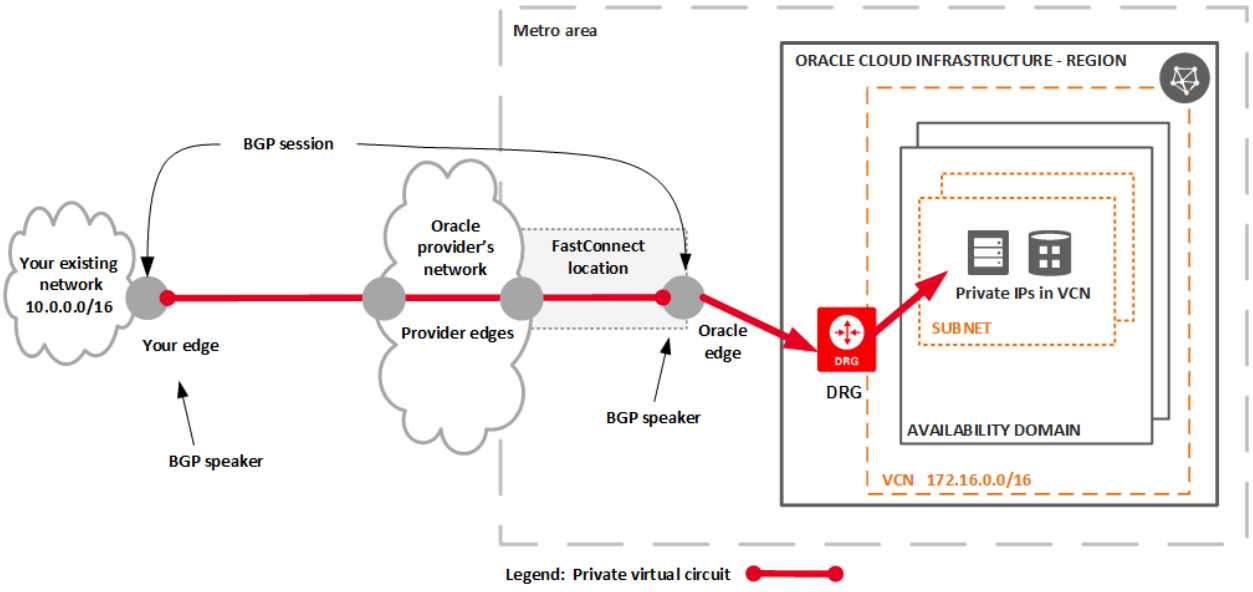
- You can add or remove public IP prefixes at any time by editing the virtual circuit. If you add a new prefix, Oracle first verifies your company's ownership before advertising it across the connection. If you remove a prefix, Oracle stops advertising the prefix within a few minutes of your editing the virtual circuit.

Oracle Provider Scenario: BGP Session to Either Oracle or the Oracle Provider

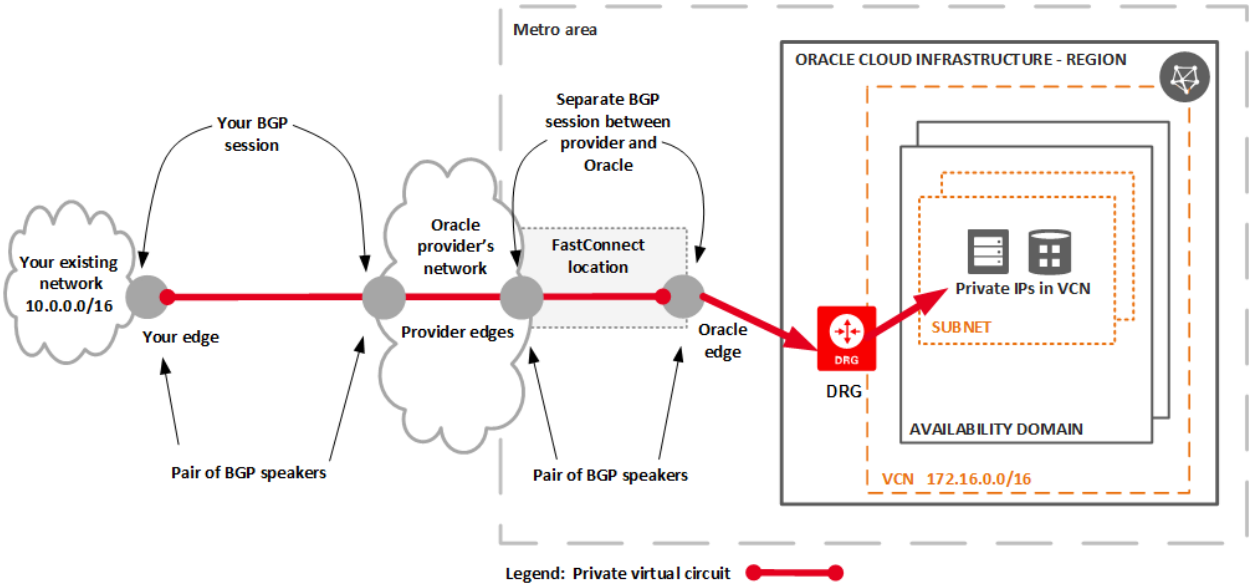
This section is applicable if you're using FastConnect through an Oracle provider. A Border Gateway Protocol (BGP) session is established from your edge, but where it goes to depends on which Oracle provider you use.

> **Tip**
>
> For simplicity, the following diagrams show only a private virtual circuit. However, the location of the BGP session is the same for a public virtual circuit.

To Oracle: With some of the Oracle providers, the BGP session goes from your edge to Oracle, as shown in the following diagram. When setting up the virtual circuit with Oracle, you are asked to provide basic BGP peering information (see General Requirements).



To the Oracle provider: With other Oracle providers, your BGP session goes from your edge to the provider's, as shown in the following diagram. When setting up the virtual circuit with Oracle, you are NOT asked for any BGP session information. Instead, you share BGP information with your Oracle provider. Notice that there's a separate BGP session that the provider establishes with Oracle.



What's Next?

See these topics to get started:

- FastConnect Requirements
- FastConnect: With an Oracle Provider
- FastConnect: With a Third-Party Provider
- FastConnect: Colocation with Oracle

FastConnect Requirements

This topic covers the requirements for implementing FastConnect.

For general information about FastConnect, see FastConnect.

Before Getting Started: Learn and Plan

Here are basic things you need to do before getting started with FastConnect:

- **FastConnect concepts:** Make sure you're familiar with the basic concepts covered in FastConnect.
- **Limits increase:** If you are colocated with Oracle, you must ask Oracle to increase your account limits for cross-connects. By default, these limits are initially set to 0, and a request to create a cross-connect will fail. For instructions, see Requesting a Service Limit Increase. In your request, make sure to indicate the region where you need the resources. It can take a couple of business days for the limit increase to take effect.
- **Hardware or routing requirements:** Review the requirements.
- **Tenancy setup and compartment design:** If you haven't yet, set up your tenancy. Think about who needs access to Oracle Cloud Infrastructure and how. For more information, see "Setting Up Your Tenancy" in the *Oracle Cloud Infrastructure Getting Started Guide*. Specifically for FastConnect, see Required IAM Policy to understand the policy required to work with FastConnect components.
- **Cloud network design:** Design your virtual cloud network (VCN), including how you want to allocate your VCN's subnets, define security list rules, define route rules, set up load balancers, and so on. For more information, see Overview of Networking.
- **Redundancy:** Think through your overall redundancy model to ensure your network can handle planned maintenance by Oracle or your organization, and unexpected failures of the various components. If you're connecting through an Oracle provider, see Network Design for Redundancy.
- **Public IP prefixes:** If you plan to set up a public virtual circuit, get the list of the public IP prefixes that you want to use with the connection. Oracle must validate your

organization's ownership of each of the prefixes before advertising each one over the connection.

- **Cloud network setup:** Set up your VCN, subnets, DRG, security lists, IAM policies, and so on, according to your design.

General Requirements

Before getting started with FastConnect, make sure you meet the following requirements:

- Oracle Cloud Infrastructure account, with at least one user with appropriate Oracle Cloud Infrastructure Identity and Access Management (IAM) permissions (for example, a user in the Administrators group).
- Network equipment that supports Layer 3 routing using BGP.
- For colocation with Oracle: Ability to connect using single mode fiber in your selected FastConnect location. Also see Hardware and Routing Requirements.
- For connection to an Oracle provider: At least one physical network connection with the provider. Also see Hardware and Routing Requirements.
- For connection to a third-party provider: At least one physical connection with the provider. Also see Hardware and Routing Requirements.
- For private peering only: At least one existing DRG set up for your VCN.
- For public peering only: The list of public IP address prefixes that you want to use with the connection. Oracle will validate your ownership of each prefix.



Important

If you're colocating with Oracle, you must ask Oracle to increase your account limits for cross-connects. By default, these limits are initially set to 0, and a request to create one of these resources will fail. For instructions, see Requesting a Service Limit Increase. In your request, make sure to **indicate the region where you need the resources**. It can take a couple of business days for the limit increase to take effect.

Hardware and Routing Requirements

If you're using an Oracle provider

Here are general routing requirements for FastConnect. These are particularly relevant if your BGP session is between your edge and Oracle.

- **IP addressing supported:** IPv4
- **P2P IP addresses:**
  - For public virtual circuits, Oracle specifies the IP addresses.
  - For private virtual circuits where your BGP session is between your edge and Oracle, you specify these addresses (/30 or /31, and one pair per virtual circuit). If you set up multiple private virtual circuits that go to the same DRG, you must use a different address on your edge router for each virtual circuit.
- **Maximum IP MTU:** 9000
- **Routing protocol:** BGPv4
- **BGP prefix limit:** For public virtual circuits: 200 prefixes. For private virtual circuits: 2000 prefixes.

- **BGP ASN:** 2-byte ASN. Public virtual circuits require a public ASN. Oracle's BGP ASN is 31898.
- **BGP keep-alive interval:** 60s
- **BGP hold-time interval:** 180s



Tip

By default, Oracle uses the default BGP timers of 60 seconds for keep-alive and 180 seconds for hold-time. If you need fast BGP convergence, you can use any value in these supported ranges: 6-60 seconds for keep-alive, and 18-180 seconds for hold-time.

If you're colocating in an FastConnect location or using a third-party provider

For the cross-connect group and cross-connects:

- **Ethernet:** 10 GbE
- **Fiber type:** 1310 NM Single Mode
- **Signal loss:** <-12 dB
- **Optics:** 10G LR
- **Fiber redundancy:** Multiple 10GE with device-level redundancy
- **Minimum capacity:** 1 x 10 GbE
- **LAG protocol:** LACP with short timers (3 @ 1s)
- **VLAN tagging:** 802.1q (single tag)
- **VLAN range:** 100-4094 (VLANs are assigned by you)
- **Maximum interface MTU:** 9196 (include 4-byte FCS trailer)

For routing:

- **IP addressing supported:** IPv4
- **P2P IP addresses:**
  - For public virtual circuits, Oracle specifies the IP addresses.
  - For private virtual circuits, you specify the addresses (a /30 or /31). You need one pair of IP addresses per private virtual circuit. If you set up multiple private virtual circuits that go to the same DRG, you must use a different address on your edge router for each virtual circuit.
- **Maximum IP MTU:** 9000
- **Routing protocol:** BGPv4
- **BGP prefix limit:** For public virtual circuits: 200 prefixes. For private virtual circuits: 2000 prefixes.
- **BGP ASN:** 2-byte ASN. Public virtual circuits require a public ASN. Oracle's BGP ASN is 31898.
- **BGP keep-alive interval:** 60s
- **BGP hold-time interval:** 180s



Tip

By default, Oracle uses the default BGP timers of 60 seconds for keep-alive and 180 seconds for hold-time. If you need fast BGP convergence, you can use any value in these supported ranges: 6-60 seconds for keep-alive, and 18-180 seconds for hold-time.

Required IAM Policy

If you're using an Oracle provider

To work with Networking resources such as dynamic routing gateways (DRGs), VCNs, and virtual circuits, you need to have a user login to the Console, and your user needs sufficient authority (by way of an IAM policy) to perform all the instructions below. If your user is in the Administrators group, you have the required authority.

If your user is not, then a policy like this would generally cover all the Networking resources:

```
Allow group NetworkAdmins to manage virtual-network-family in tenancy
```

To *only* create and manage a virtual circuit, you would need a policy like this:

```
Allow group VirtualCircuitAdmins to manage drgs in tenancy

Allow group VirtualCircuitAdmins to manage virtual-circuits in tenancy
```

The first statement (to manage DRGs) is necessary only for private virtual circuits.

For more information, see Getting Started with Policies and Common Policies.

If you're colocating in a FastConnect location or using a third-party provider

To work with Networking resources such as dynamic routing gateways (DRGs), VCNs, virtual circuits, and cross-connects, you need to have a user login to the Console, and your user needs sufficient authority (by way of an IAM policy) to perform all the instructions that follow. If your user is in the Administrators group, you have the required authority.

If your user is not, then a policy like this would generally cover all the Networking resources:

```
Allow group NetworkAdmins to manage virtual-network-family in tenancy
```

To *only* create and manage cross-connects, cross-connect groups, and virtual circuits, you would need a policy like this:

```
Allow group FastConnectAdmins to manage drgs in tenancy

Allow group FastConnectAdmins to manage cross-connects in tenancy

Allow group FastConnectAdmins to manage cross-connect-groups in tenancy

Allow group FastConnectAdmins to manage virtual-circuits in tenancy
```

The first statement (to manage DRGs) is necessary only for private virtual circuits.

For more information, see Getting Started with Policies and Common Policies.

What's Next?

Choose the topic that suits your situation:

- FastConnect: With an Oracle Provider
- FastConnect: With a Third-Party Provider
- FastConnect: Colocation with Oracle

FastConnect: With an Oracle Provider

This topic is for customers who want to use Oracle Cloud Infrastructure FastConnect by connecting to an Oracle provider. For a summary of the different ways to connect, see the connectivity models.

If you instead want to use a network provider that is not on the list of Oracle providers, see FastConnect: With a Third-Party Provider. Or if you want to use FastConnect by colocating with Oracle, see FastConnect: Colocation with Oracle.

For general information about FastConnect, see FastConnect.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Network Design for Redundancy

This section gives guidelines on how to design your network for redundancy so that it meets requirements for the Oracle Cloud Infrastructure FastConnect Service-Level Agreement (SLA).

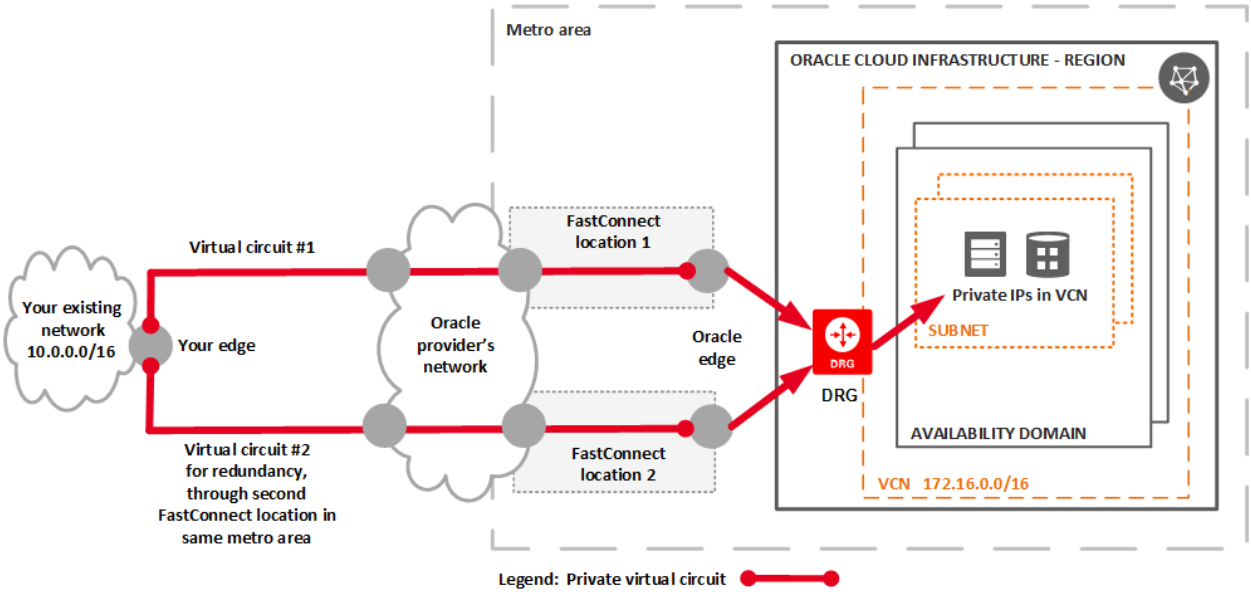
In general, you should design your network with both of these in mind:

- Regularly scheduled maintenance by Oracle, your provider, or your own organization.
- Unexpected failures on the part of Oracle, your provider, or you own networking components. Failures are rare but need to be planned for.

For redundancy, Oracle provides:

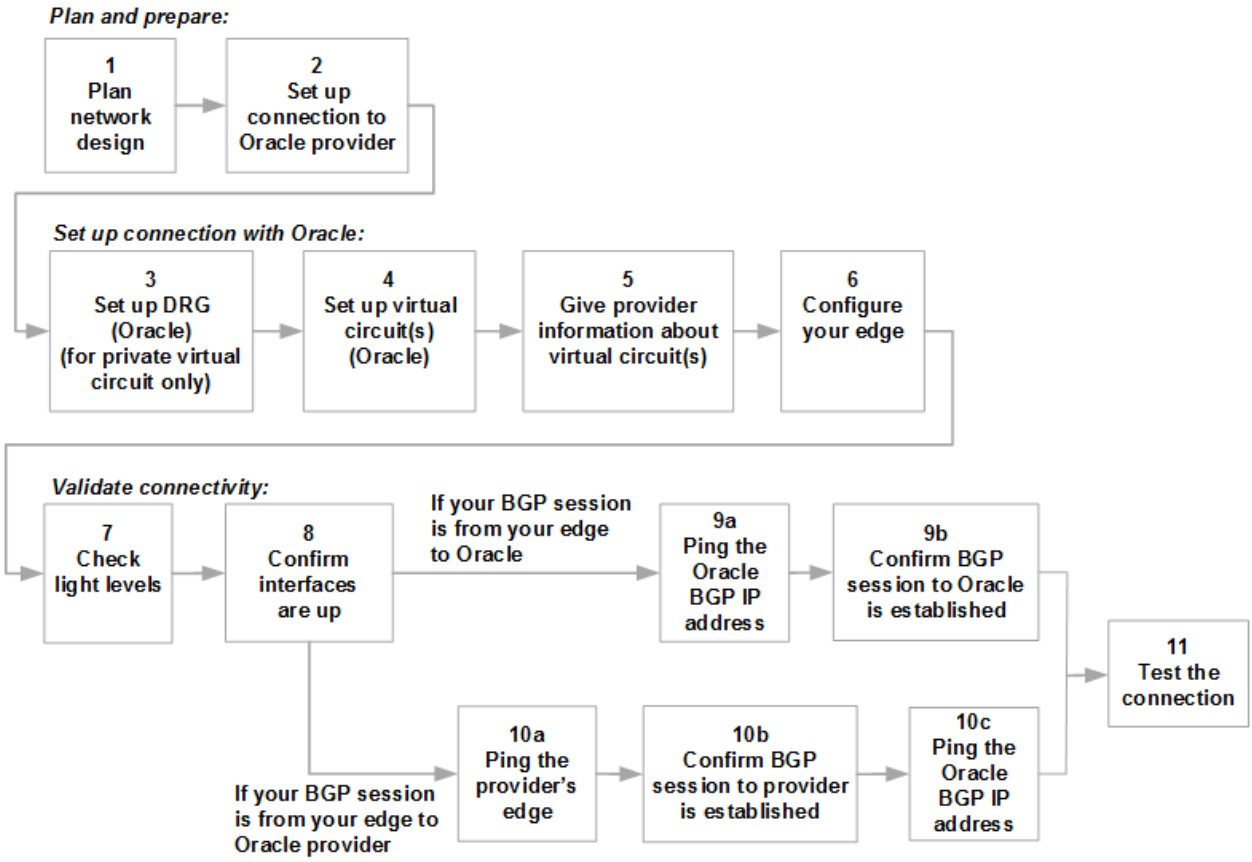
- Multiple FastConnect locations within each metro area
- Multiple routers in each FastConnect location
- Multiple physical circuits in each FastConnect location

Oracle handles redundancy of the routers and physical circuits in the FastConnect locations. You should then handle redundancy of the physical connection between your existing network and the Oracle provider. To do that, create two virtual circuits. Ensure that each runs on a different physical network connection to the provider, and goes to a different FastConnect location in the same metro area. Both virtual circuits go to the same DRG (if they're private virtual circuits). You'll have two separate BGP sessions from your edge to Oracle (one per virtual circuit). See the following diagram. An active/active configuration for routing traffic across the two connections is recommended and supported by Oracle.


Metro area
ORACLE CLOUD INFRASTRUCTURE - REGION
Virtual circuit #1
Your existing network 10.0.0.0/16
Your edge
Oracle provider's network
FastConnect location 1
Oracle edge
DRG
DRG
Private IPs in VCN
SUBNET
AVAILABILITY DOMAIN
VCN 172.16.0.0/16
Virtual circuit #2 for redundancy, through second FastConnect location in same metro area
FastConnect location 2
Legend: Private virtual circuit

Getting Started with FastConnect

The following flow chart shows the overall process of setting up FastConnect.



Also see the sequence diagram in To get the status of your virtual circuit.

Task 1: Learn and plan

If you haven't yet, walk through the planning in Before Getting Started: Learn and Plan. Also

see Network Design for Redundancy.

Task 2: Set up connection to the Oracle provider

If you haven't already, start the process of ordering the connection from the Oracle provider, setting it up, and then testing it with the provider. It can take some time, depending on the provider.

Task 3: Set up a DRG (private peering only)

Summary: If you plan to use a private virtual circuit, you need a DRG. If you haven't already, use the Oracle Cloud Infrastructure Console to set up a DRG, attach it to your VCN, and update routing in your VCN to include a route rule to send traffic to the DRG. It's easy to forget to update the route table. Without the route rule, no traffic flows.

Instructions:

- To create a dynamic routing gateway
- To attach a dynamic routing gateway to a VCN
- To update rules in an existing route table

Task 4: Set up your virtual circuit(s)

Summary: Create a virtual circuit in the Oracle Console. If your network design includes more than one virtual circuit, complete the following steps for each one.

Instructions:

Repeat the following steps for each virtual circuit you want to create.

1. In the Console, confirm you're viewing the compartment that you want to work in. If you're not sure which one, use the compartment that contains the DRG that you'll connect to (for a private virtual circuit). This choice of compartment, in conjunction with

a corresponding IAM policy, controls who has access to the virtual circuit you're about to create.

2. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **FastConnect**.
   The resulting **FastConnect** page is where you'll create a new connection and later return to when you need to manage the connection.
3. Click **Create Connection**.
4. Select **Connect via provider** and choose the provider from the list.
   The resulting dialog box shows the information required to set up the virtual circuit.
5. Enter the following for your virtual circuit:
   - **Name:** A friendly name that helps you keep track of your virtual circuits. The value does not need to be unique across your virtual circuits, and you can change it later. Avoid entering confidential information.
   - **Create in Compartment:** Leave as is (the compartment you're currently working in).
6. Choose the virtual circuit type (private or public). A private virtual circuit is for private peering (where your existing network receives your VCN's private IP addresses). A public virtual circuit is for public peering (where your existing network receives the Oracle Cloud Infrastructure regional public IP addresses). Also see Uses for FastConnect.
   - For a private virtual circuit, enter the following:
     - **Dynamic Routing Gateway Compartment:** Select the compartment where the DRG resides (it should already be selected).
     - **Dynamic Routing Gateway:** Select the DRG to route the FastConnect traffic to.
     - **Provisioned Bandwidth:** Choose your desired value. If your bandwidth needs increase later, you can update the virtual circuit to use a different value (see To edit a virtual circuit).

If your BGP session goes to Oracle (see Oracle Provider Scenario: BGP Session to Either Oracle or the Oracle Provider), the dialog box includes additional fields for the BGP session:

- **Customer BGP IP Address:** The BGP peering IP address for your edge router, with either a /30 or /31 subnet mask.
- **Oracle BGP IP Address:** The BGP peering IP address you want to use for the DRG, with either a /30 or /31 subnet mask.
- **Customer BGP ASN:** The public or private ASN for your network.

- For a public virtual circuit, enter the following:
  - **Provisioned Bandwidth:** Choose your desired value. If your bandwidth needs increase later, you can update the virtual circuit to use a different value (see To edit a virtual circuit).
  - **Customer BGP ASN:** The public ASN for your network. Present only if your BGP session goes to Oracle (see Oracle Provider Scenario: BGP Session to Either Oracle or the Oracle Provider). Note that Oracle specifies the BGP IP addresses for a public virtual circuit.
  - **Public IP Prefixes:** The public IP prefixes that you want Oracle to receive over the connection (each one must be /31 or less specific). You can enter a comma-separated list of prefixes, or one per line.

7. Click **Continue**.
   The virtual circuit is created. Its OCID and a link to the provider's portal are displayed in the resulting confirmation dialog box. The OCID is also available on the main FastConnect page and with the other virtual circuit details.
8. Copy and paste the OCID to another location. You give it to your provider in the next task.
9. Click **Close**.

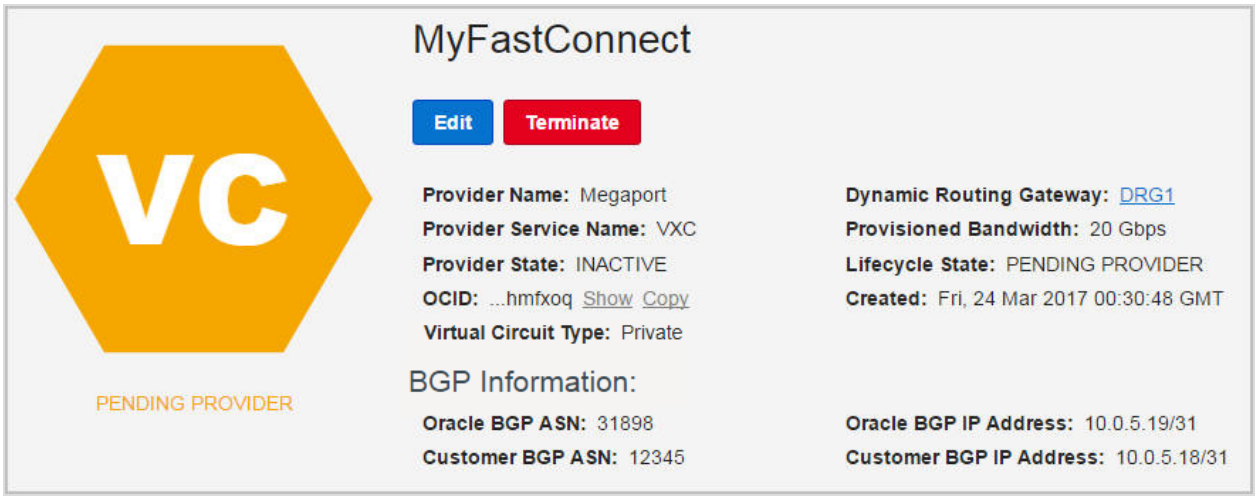
The virtual circuit is listed on the FastConnect page. You can click the virtual circuit to see the full set of details. These items indicate the status of the connection:

- **Provider State:** Whether the provider is aware of your request to create a virtual circuit and is provisioning it from their end. At this point, the value is INACTIVE.
- **Lifecycle State:** The current status of the virtual circuit *during the time it's being set up*. At this point, the value is PENDING_PROVIDER.
- **Large "VC" icon:** While the virtual circuit is still being set up, this large, colored icon also indicates the Lifecycle State (for example, PENDING_PROVIDER). After the Lifecycle State switches to PROVISIONED, this icon switches to indicate the state of the virtual circuit's BGP session (either green/UP or red/DOWN).



Tip

For a virtual circuit where your BGP session goes to the Oracle provider, the BGP session state icon for the virtual circuit reflects the status of the *separate BGP session between the Oracle provider and Oracle*. For reference, see Oracle Provider Scenario: BGP Session to Either Oracle or the Oracle Provider.



Also see the diagram in To get the status of your virtual circuit.

Task 5: Give the provider information about the virtual circuit(s)

Summary: Provide the OCID of each virtual circuit you created, along with any other information the provider requests. The provider configures each virtual circuit on their end to complete the connectivity.

Instructions for AT&T

You don't need to provide AT&T the OCID of your virtual circuit. Instead, AT&T and Oracle use other information to coordinate the provisioning of the virtual circuit. Follow these steps:

1. Work with your AT&T account team to sign up for NetBond for Cloud services.
   In return, you receive a welcome letter with credentials for the AT&T Cloud Services Portal.
2. Sign in to the AT&T Cloud Services portal and create one Virtual Network Connection (VNC). You must provide these items: name of AT&T MPLS VPN, region, free-form name for the VNC, and a minimum bandwidth commitment.

3. Inside the VNC, create a VLAN. You must provide a /29 address space and free-form name.

   In return, you receive a Service Key for AT&T NetBond for Cloud.
4. Give Oracle the Service Key you received in the preceding step. To do this, create a service request at My Oracle Support.

Oracle uses the Service Key that you provided to provision the virtual circuit(s). The process takes typically 1-2 business days. During that time, the virtual circuit's **Provider State** changes to ACTIVE, and the **Lifecycle State** changes to PROVISIONING. When the virtual circuit is completely set up, the **Lifecycle State** switches to PROVISIONED.

Instructions for BT Cloud Connect

1. Contact your BT account manager to order Oracle Cloud Infrastructure FastConnect.
2. Provide the OCID of your virtual circuit.

BT contacts Oracle, and together they provision the virtual circuit. During that time, the virtual circuit's **Provider State** changes to ACTIVE, and the **Lifecycle State** changes to PROVISIONING. When the virtual circuit is completely set up, the **Lifecycle State** switches to PROVISIONED.

Instructions for Equinix Cloud Exchange

1. Go to the Equinix Cloud Exchange portal and sign in.
2. Click the **Create Connection** tab and select the following:
   - **Metro**: Your desired location (for example, Ashburn).
   - **Service:** Oracle FastConnect - Oracle Cloud Infrastructure (Layer 2 Connection)
3. In the **Buyer-side VLAN and Port** section, enter a friendly name for the connection, the physical connection you want to use (the Buyer-side Port), and the VLAN ID.

4. In the **Additional Buyer-side Information** section, enter your ASN and the IP address you specified for Oracle's side of the BGP peering session.
5. In the **Seller-side Information** section, enter the OCID for the virtual circuit.
6. Choose a speed for the connection.
7. Enter the requested email address and click **Create Connections**.

If your network design includes a second physical connection and virtual circuit for redundancy, repeat the steps above with the second port you've set up with Equinix Cloud exchange and the second virtual circuit.

Oracle receives an email and then provisions the virtual circuit(s). The process takes typically 1-2 business days. During that time, the virtual circuit's **Provider State** changes to ACTIVE, and the **Lifecycle State** changes to PROVISIONING. When the virtual circuit is completely set up, the **Lifecycle State** switches to PROVISIONED.

Instructions for Interxion CloudConnect

1. Go to the Interxion portal and sign in.
2. Create a new **Cloud Connect**.
3. Fill in the requested contact information.
4. For the **Cloud Service Details**, enter the following:
   - **Cloud Service Provider**: Oracle
   - **Cloud Service Offer**: Oracle FastConnect
   - **Cloud Service Location**: Where you're connecting
   - **Bandwidth:** Your desired bandwidth
   - **OCID:** The OCID for the virtual circuit
5. Select a Cloud Access Port to use for the virtual circuit, and fill in the details for your Cloud Access Port.
6. Accept and confirm your order. You'll get a confirmation email.

If your network design includes a second Cloud Access Port and virtual circuit for redundancy, repeat the preceding steps with the second Cloud Access Port you've set up with Interxion and the second virtual circuit.

On the Oracle Console, you will soon see the virtual circuit's **Provider State** change to ACTIVE. The **Lifecycle State** will also change to PROVISIONING. Oracle's system will then complete the virtual circuit setup, and the **Lifecycle State** will shortly switch to PROVISIONED. For more information, see the diagram in To get the status of your virtual circuit.

Instructions for Megaport

1. Go to the Megaport console and sign in.
2. Locate your existing Megaport and click **+ Connection** to add a connection. If you haven't begun the process of setting up a Megaport yet, you need to start that first and then add the connection to it.
3. Click **VXC to Cloud** and click the icon for Oracle.
4. Enter the OCID for the virtual circuit and choose which Oracle CloudTarget Port (that is, which building) to use for the virtual circuit.

If your network design includes a second Megaport and virtual circuit for redundancy, repeat the preceding steps with the second Megaport you've set up and the second virtual circuit. Make sure to choose the other building when specifying the Oracle CloudTarget Port for the virtual circuit.

On the Oracle Console, you will soon see the virtual circuit's **Provider State** change to ACTIVE. The **Lifecycle State** will also change to PROVISIONING. Oracle's system will then complete the virtual circuit setup, and the **Lifecycle State** will shortly switch to PROVISIONED. For more information, see the diagram in To get the status of your virtual circuit.

Instructions for Orange Business Services Business VPN Galerie

1. Contact your Orange Business Services account manager to order Oracle Cloud Infrastructure FastConnect.
2. Provide the OCID of your virtual circuit.

Orange Business Services contacts Oracle, and together they provision the virtual circuit. During that time, the virtual circuit's **Provider State** changes to ACTIVE, and the **Lifecycle State** changes to PROVISIONING. When the virtual circuit is completely set up, the **Lifecycle State** switches to PROVISIONED.

Instructions for QTS

1. Contact your QTS sales representative to order Oracle Cloud Infrastructure FastConnect. QTS might ask for your desired bandwidth (the physical port speed and FastConnect rate limit).
2. Provide the OCID of your virtual circuit. If your network design includes a second virtual circuit for redundancy, also provide the OCID for the second virtual circuit.

On the Oracle Console, you will soon see the virtual circuit's **Provider State** change to ACTIVE. The **Lifecycle State** will also change to PROVISIONING. Oracle's system will then complete the virtual circuit setup, and the **Lifecycle State** will shortly switch to PROVISIONED. For more information, see the diagram in To get the status of your virtual circuit.

Instructions for Sohonet

1. Contact your Sohonet account manager to order the FastLane service.
2. Provide the OCID of your virtual circuit.

Sohonet then sends you an order acknowledgment, provisions the virtual circuit, and confirms the virtual circuit's activation with you.

Instructions for Tata IZO Private Connect

1. Contact your Tata account manager to order Oracle Cloud Infrastructure FastConnect.
2. Provide the OCID of your virtual circuit.

Tata contacts Oracle, and together they provision the virtual circuit. During that time, the virtual circuit's **Provider State** changes to ACTIVE, and the **Lifecycle State** changes to PROVISIONING. When the virtual circuit is completely set up, the **Lifecycle State** switches to PROVISIONED.

Instructions for Verizon SCI

1. Contact your Verizon account manager to order Verizon SCI Connectivity for Oracle Cloud Infrastructure: FastConnect. If you need help finding your Verizon account manager, see http://www.verizonenterprise.com/Support/contact.
2. Provide the following information to Verizon:
   - The location where Oracle FastConnect is provisioned (for example, IAD)
   - The OCID for your virtual circuit
   - *FastConnect OCI: Private Peering* as the SCI Partner Service Description

   Verizon then works with you to determine the best time to activate the service. At the designated time, your Verizon account manager provisions the service with the OCID of your virtual circuit and verifies the service has been provisioned. You then receive an email from Verizon with network configuration information.
3. If setting up FastConnect in the uk-london-1: Give Oracle the network configuration information you received. To do this, create a service request at My Oracle Support.

Oracle uses the network configuration information to provision the virtual circuit(s). The process takes typically 1-2 business days. During that time, the virtual circuit's **Provider State** changes to ACTIVE, and the **Lifecycle State** changes to PROVISIONING. When the virtual circuit is completely set up, the **Lifecycle State** switches to PROVISIONED.

Instructions for Windstream

1. Contact your Windstream sales representative to order Oracle Cloud Infrastructure FastConnect. Windstream might ask for your desired bandwidth (the physical port speed and FastConnect rate limit).
2. Provide the OCID of your virtual circuit. If your network design includes a second virtual circuit for redundancy, also provide the OCID for the second virtual circuit.

Oracle receives an email and then provisions the virtual circuit. The process takes typically 1-2 business days. During that time, the virtual circuit's **Provider State** changes to ACTIVE, and the **Lifecycle State** changes to PROVISIONING. When the virtual circuit is completely set up, the **Lifecycle State** switches to PROVISIONED.

Instructions for Zayo

1. Contact your Zayo account manager to order Oracle Cloud Infrastructure FastConnect through Tranzact by Zayo.
2. Provide the OCID of your virtual circuit.

Zayo then sends you an order acknowledgment, provisions the virtual circuit, and confirms the virtual circuit's activation with you.

Task 6: Configure your edge

If your BGP session goes to Oracle: (see Oracle Provider Scenario: BGP Session to Either Oracle or the Oracle Provider), configure your edge router(s) to use the BGP peering information (see General Requirements). Oracle's BGP ASN is 31898. By default, Oracle uses the default BGP timers of 60 seconds for keep-alive and 180 seconds for hold-time. If you need fast BGP convergence, you can use any value in these supported ranges: 6-60 seconds for keep-alive, and 18-180 seconds for hold-time. Also configure the router(s) for redundancy according to the network design you decided on earlier (see Network Design for Redundancy). After you successfully configure the BGP session, the virtual circuit's BGP session state

switches to UP.

If your BGP session instead goes to the Oracle provider: You still need to configure your router(s) if you haven't already. You may need to contact your provider to get the required BGP peering information. This BGP session must be up and running for FastConnect to work. Also configure your edge router(s) for redundancy according to the network design you decided on earlier (see Network Design for Redundancy).



Important

If this is a public virtual circuit, Oracle's public IP addresses are advertised over both FastConnect and your connection with your Internet Service Provider (ISP). Make sure to give higher preference to FastConnect over your ISP.

Task 7: Check light levels

Confirm the light levels are good for each of your physical network connections to the provider. Don't proceed until they are.

Task 8: Confirm your interfaces are up

Confirm your side of the interfaces for the connections to the provider are up. Don't proceed until they are.

BGP Session Goes to Oracle

Task 9a: Ping the Oracle BGP IP address

For each virtual circuit, ping the Oracle BGP IP address assigned to the virtual circuit. Check the error counters and look for any dropped packets. Don't proceed until you can successfully

ping this IP address without errors.

Task 9b: Confirm the BGP session is established

For each virtual circuit, confirm the BGP session is in an established state. When it is, the connection is ready to test (see Task 11: Test the connection).

BGP Session Goes to the Provider

Task 10a: Ping the provider's edge

For each virtual circuit, ping the provider's edge. Check the error counters and look for any dropped packets. Don't proceed until you can successfully ping the provider's edge without errors.

Task 10b: Confirm the BGP session is established

Confirm the BGP session you have with the provider is in an established state. Don't proceed until it is.

Task 10c: Ping the Oracle BGP IP address

For each virtual circuit, ping the Oracle BGP IP address (which you can get from the provider). Check the error counters and look for any dropped packets. When you can successfully ping this IP address without errors, the connection is ready to test.

Task 11: Test the connection

For a private virtual circuit: You should be able to launch an instance in your VCN and access it (for example, with SSH) from a host in your existing private network. See Creating an Instance. If you can, your FastConnect private virtual circuit is ready to use.

For a public virtual circuit:

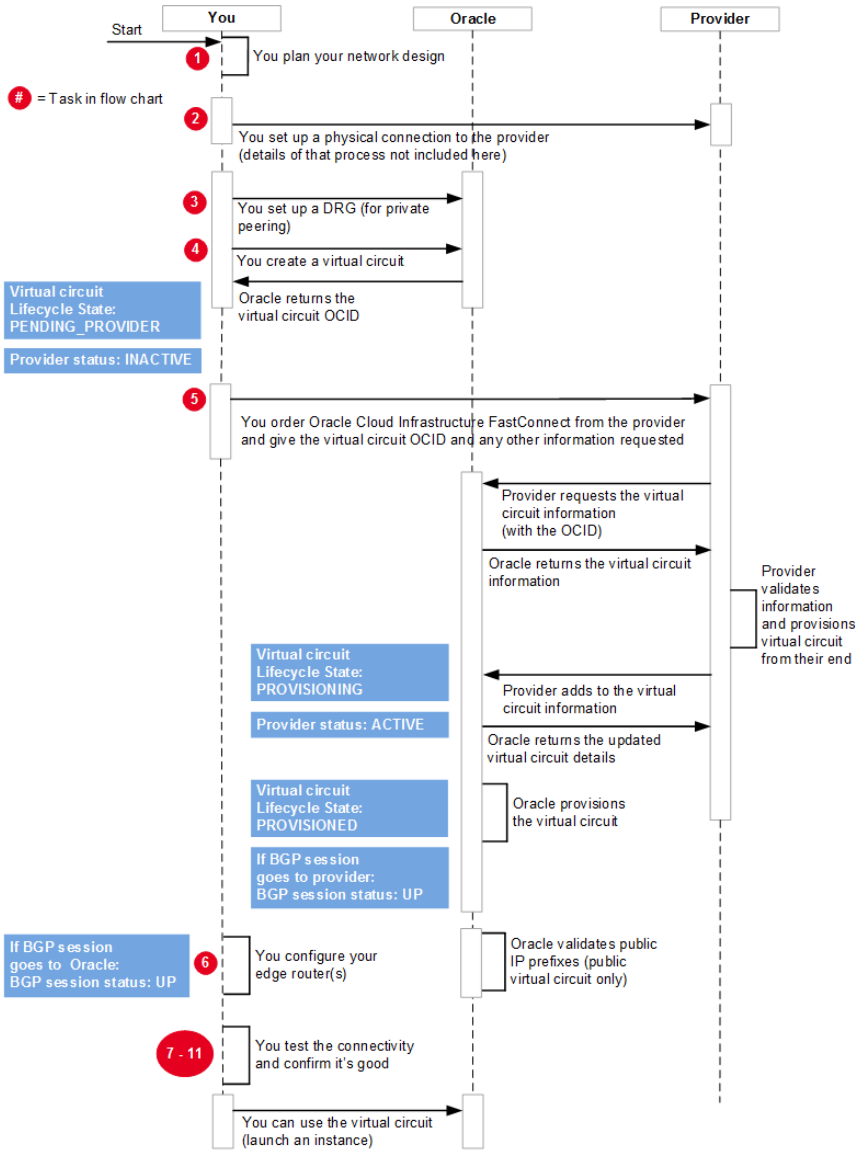
1. Make sure that Oracle has successfully verified *at least one* of the public prefixes you've submitted. You can see the status of each prefix by viewing the virtual circuit's details in the Console. When one of the prefixes has been validated, Oracle starts advertising the regional OCI public addresses over the connection.
2. Launch an instance with a public IP address.
3. Ping the public IP address from a host in your existing private network. You should see the packet on the FastConnect interface on the virtual circuit. If you do, your FastConnect public virtual circuit is ready to use. However, remember that *only the public prefixes that Oracle has successfully verified so far* are advertised on the connection.

Managing Your Virtual Circuit

To get the status of your virtual circuit

1. In the Console, go to **Networking**, and then click **FastConnect** to view your list of connections.
2. Click the virtual circuit you're interested in to view its details.

The following diagram shows the different states of the virtual circuit when you're setting it up.


You
Oracle
Provider
Start
1
You plan your network design
= Task in flow chart
2
You set up a physical connection to the provider (details of that process not included here)
3
You set up a DRG (for private peering)
4
You create a virtual circuit
Oracle returns the virtual circuit OCID
Virtual circuit Lifecycle State: PENDING_PROVIDER
Provider status: INACTIVE
5
You order Oracle Cloud Infrastructure FastConnect from the provider and give the virtual circuit OCID and any other information requested
Provider requests the virtual circuit information (with the OCID)
Oracle returns the virtual circuit information
Provider validates information and provisions virtual circuit from their end
Virtual circuit Lifecycle State: PROVISIONING
Provider adds to the virtual circuit information
Provider status: ACTIVE
Oracle returns the updated virtual circuit details
Virtual circuit Lifecycle State: PROVISIONED
Oracle provisions the virtual circuit
If BGP session goes to provider: BGP session status: UP
If BGP session goes to Oracle: BGP session status: UP
6
You configure your edge router(s)
Oracle validates public IP prefixes (public virtual circuit only)
7 - 11
You test the connectivity and confirm it's good
You can use the virtual circuit (launch an instance)

To edit a virtual circuit

You can change these items for a virtual circuit:

- The name
- The bandwidth
- Which DRG it uses (for a private virtual circuit)
- The public IP prefixes (for a public virtual circuit)
- Depending on the situation, you might also have access to the BGP session information for the virtual circuit and thus can change it.



Important

If your virtual circuit is working and in the PROVISIONED state before you edit it, be aware that changing any of the properties besides the name, bandwidth, and public prefixes (for a public virtual circuit) causes the virtual circuit's state to switch to PROVISIONING and **may cause the related BGP session to go down.** After Oracle re-provisions the virtual circuit, its state returns to PROVISIONED. Make sure you confirm that the associated BGP session is back up.

If you change the public IP prefixes for a public virtual circuit, the BGP session and virtual circuit status is unaffected. However, the overall FastConnect status may change to IP CHECK IN PROGRESS or IP CHECK FAILED. Oracle begins advertising a new IP prefix over the connection only after verifying your ownership of it.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **FastConnect**.
2. Select the compartment where the connection resides, and then click the connection to view its details.
3. Click the virtual circuit to view its details.
4. Click **Edit** and make your changes.
5. Click **Save**.

To terminate a virtual circuit



Important

Also terminate the connection with the provider, or else the provider may continue to bill you.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **FastConnect**.
2. Select the compartment where the connection resides, and then click the connection to view its details.
3. Click the virtual circuit to view its details.
4. Click **Delete**.
5. Confirm when prompted.

The virtual circuit's Lifecycle State changes to TERMINATING and then to TERMINATED.

To manage public IP prefixes for a public virtual circuit

For general information about the prefixes, see Logical Connection: Public Virtual Circuit.

You can specify your public IP prefixes when you create the virtual circuit. See Task 4: Set up your virtual circuit(s).

You can add or remove public IP prefixes later after creating the virtual circuit. See To edit a virtual circuit. If you add a new prefix, Oracle first verifies your company's ownership before advertising it across the connection. If you remove a prefix, Oracle stops advertising the prefix within a few minutes of your editing the virtual circuit.

You can view the state of Oracle's verification of a given public prefix by viewing the virtual circuit's details in the Console. Here are the possible values:

- IP CHECK IN PROGRESS: Oracle is in the process of verifying your organization's ownership of the prefix.
- IP CHECK FAILED: Oracle could not verify your organization's ownership. Oracle will not advertise the prefix over the virtual circuit.
- IP CHECK COMPLETED: Oracle successfully verified your organization's ownership. Oracle is advertising the prefix over the virtual circuit.

FastConnect: With a Third-Party Provider

This topic is for customers who want to use Oracle Cloud Infrastructure FastConnect by connecting to a *third-party network provider* of their choice, and not an Oracle provider. For a summary of the different ways to connect, see the connectivity models.

If you are using an Oracle provider, see FastConnect: With an Oracle Provider. Or, if you want to use FastConnect by colocating with Oracle, see FastConnect: Colocation with Oracle.

For general information about FastConnect, see FastConnect.



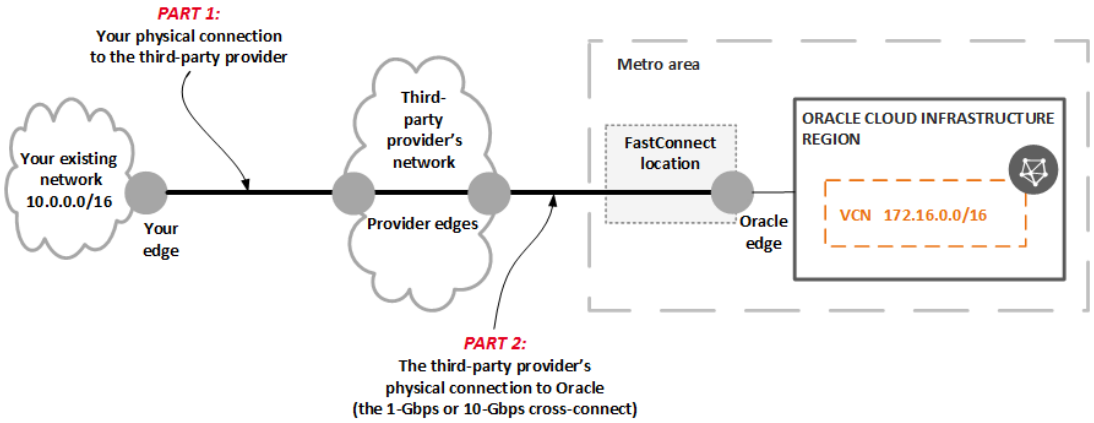
Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Important Points and Responsibilities

- You can use FastConnect by working with a third-party network service provider or carrier of your choice. The network provider must be capable of connecting to the Oracle routers in one of the FastConnect data center locations over single-mode fiber. For more detailed technical requirements, see Hardware and Routing Requirements.
- Your overall connection with the third-party provider includes two parts, as illustrated in the following diagram:
    - **Part 1:** Your physical connection to the third-party provider. The rest of this topic assumes you've already set up this part of the overall connection.
    - **Part 2:** The physical cross-connect that the third-party provider sets up in the

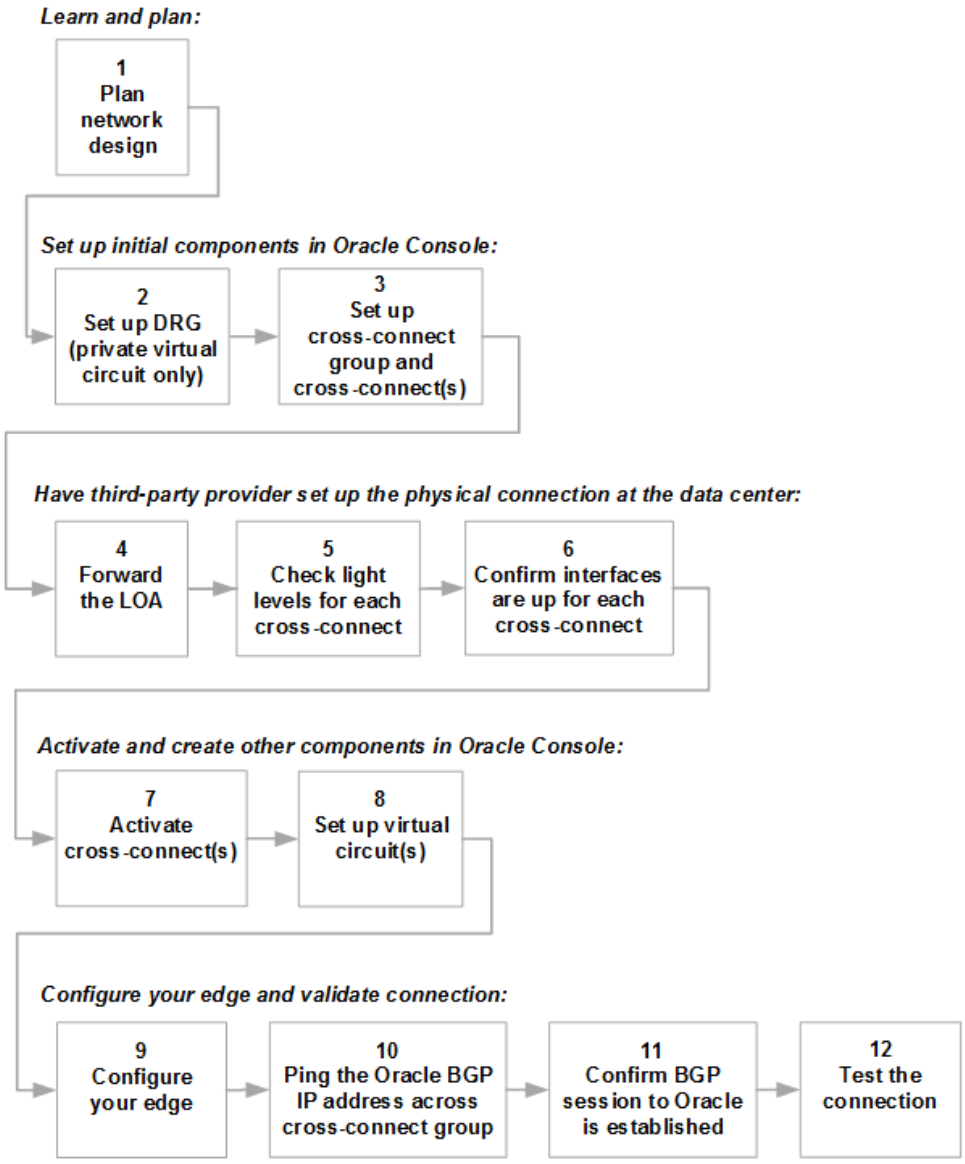
FastConnect location data center on your behalf.



- To obtain the Letter of Authorization (LOA) for the cross-connect, you must use the Oracle Console to set up a cross-connect or cross-connect group. The resulting LOA from Oracle covers part 2 of the connection in the preceding diagram.
- You must forward the LOA to your third-party provider, who is responsible for working with the data center to set up the physical cross-connect on your behalf.
- The third-party provider issues a cross-connect order with the data center facility to run fiber optics to complete the connection from the third-party provider's cage to Oracle's patch panel as described in the LOA. Typically the data center colocation staff are the ones who run the fiber optics to complete the connection.
- Each LOA is valid for only a limited time. If the physical cross-connect is not set up before the LOA's expiration, the LOA is revoked.
- **The third-party provider is responsible for charging you for the *entire* connection (both parts 1 and 2)**. Oracle does not set up this cross-connect in the data center, does not pay for it, and does not include it in your FastConnect charges.
- The LOA specifies an Oracle demarcation point. If your network provider is located at a different demarcation point in the data center cage, they must set up the cross-connect from their demarcation point to the Oracle demarcation point.

Getting Started with FastConnect

The following flow chart shows the overall process of setting up FastConnect.



Task 1: Learn and plan

If you haven't yet, walk through the planning in Before Getting Started: Learn and Plan.

Task 2: Set up a DRG (private peering only)

Summary: If you plan to use a private virtual circuit, you need a DRG. If you haven't already, use the Oracle Cloud Infrastructure Console to set up a DRG, attach it to your VCN, and update routing in your VCN to include a route rule to send traffic to the DRG. It's easy to forget to update the route table. Without the route rule, no traffic will flow.

Instructions:

- To create a dynamic routing gateway
- To attach a dynamic routing gateway to a VCN
- To update rules in an existing route table

Task 3: Set up your cross-connect group and cross-connect(s)

Summary: Create a connection in the Console, which consists of a cross-connect group that contains at least one cross-connect.

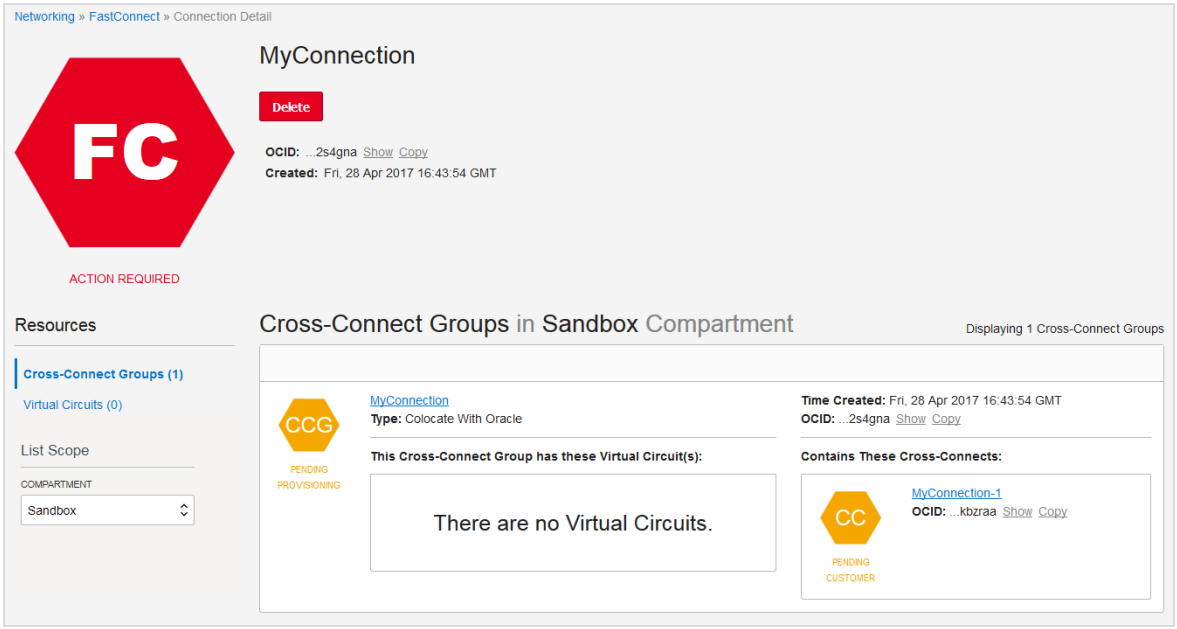
Instructions:

1. In the Console, confirm you're viewing the compartment that you want to work in. If you're not sure which one, use the compartment that contains the DRG that you'll connect to (for a private virtual circuit). This choice of compartment, in conjunction with a corresponding IAM policy, controls who has access to the cross-connect group and cross-connect(s) you're about to create.
2. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **FastConnect**.
   The resulting **FastConnect** page is where you'll create a new connection and later return to when you need to manage the connection and its components.

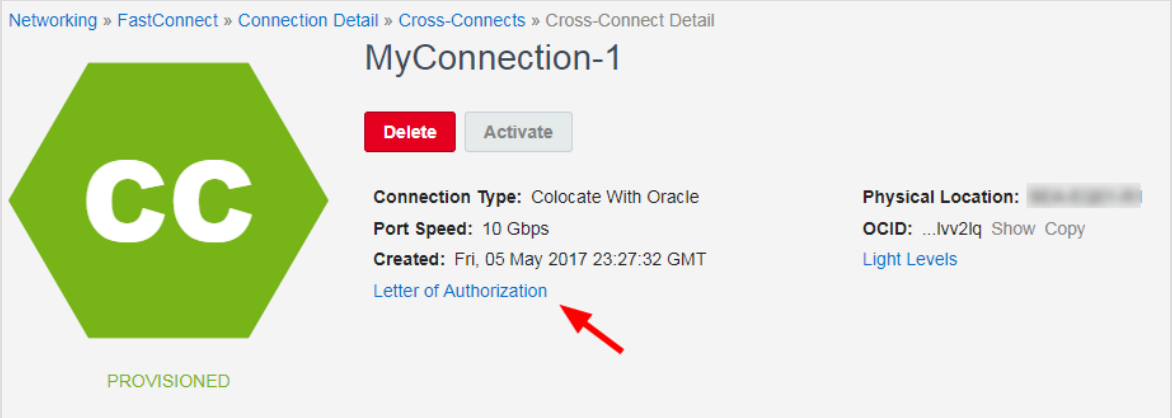
3. Click **Create Connection**.
4. Select the radio button for **Colocate with Oracle** and click **Continue**. Select this option even though a third-party provider will set up the physical connection to Oracle in the FastConnect location.

   In the resulting dialog box, you'll create a cross-connect group with up to three cross-connects in it. If you need more cross-connects in the group, you can add them later. You can have a maximum of eight cross-connects in a group.
5. Enter the following:
   - **Name:** A friendly name that helps you keep track of this connection. The cross-connect group will use this name. Each cross-connect in this group will also use this name, but with a hyphen and number appended (for example, MyName-1, MyName-2, and so on). You can't change the name later. Avoid entering confidential information.
   - **Create in Compartment:** Leave as is (the compartment you're currently working in).
   - **Number of cross-connects:** Select the number of individual cross-connects to create in the cross-connect group. In the Console, you can create three. If you need more, you can add more cross-connects later (total eight in a cross-connect group).
   - **Port speed:** All cross-connects must use a 10 Gbps port speed.
   - **Physical location:** Select the FastConnect location for this cross-connect group.
   - **Cross-Connect Group Proximity**: Appears only if you have one or more existing cross-connect groups in this FastConnect location. Here you may optionally specify whether you want the new cross-connect group to be on the same or different router than one of your other cross-connect groups.

6. Click **Continue**.
   The new connection is created and listed on the FastConnect page. Click it to see the connection details, which includes the status of each of the components (the cross-connect group, cross-connect(s), and later the virtual circuit(s)). The following screenshot shows the connection details:



7. Click each cross-connect to view its details (MyConnection-1 in the preceding screenshot), and then view and print the cross-connect's Letter of Authorization (LOA). You need to submit it with your cabling request at the FastConnect location in the next step.



Summary of status icons:

- **FastConnect (FC) icon:** The large icon in the top-left corner. It shows the general status of the overall FastConnect connection and whether you need to take action. At this point, the FC status will be ACTION REQUIRED (see the next task).
- **Cross-connect group (CCG) icon:** The icon near the middle of the page. It shows the status of the cross-connect group itself. At this point, the CCG status will be PENDING PROVISIONING.
- **Cross-connect (CC) icon:** The icon on the right side of the page. It shows the status of a given cross-connect. At this point, the CC status will be PENDING CUSTOMER.

Later, when you add a virtual circuit to your provisioned cross-connect group, under the CCG icon there will be a VC icon that shows the status of the virtual circuit.

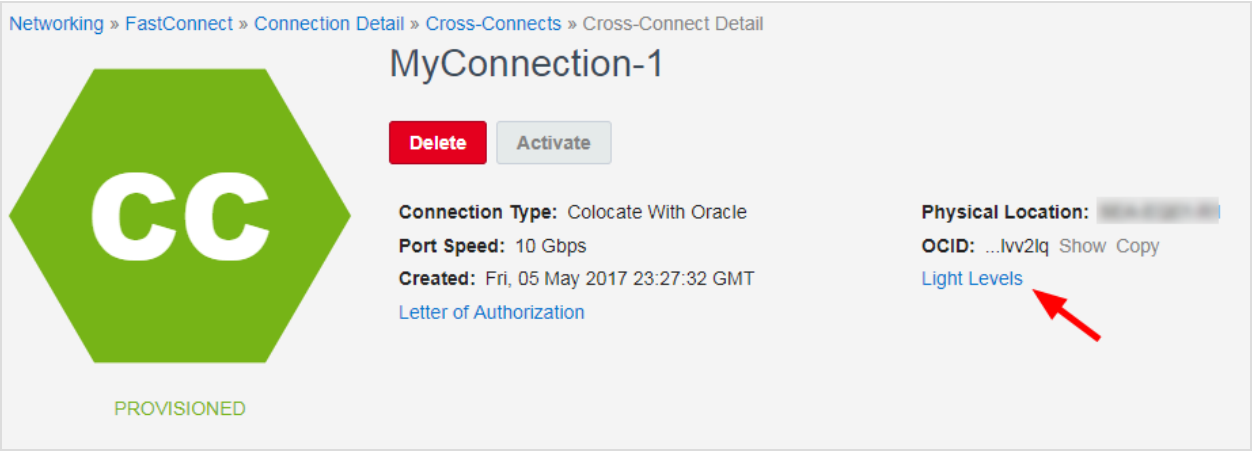
Task 4: Forward the LOA to your third-party provider

Forward the LOA or LOAs from the preceding task to your third-party network provider so they can request cabling at the FastConnect location. Each LOA is valid for a limited time. The details are printed on the LOA.

Task 5: Check light levels

After the third-party provider completes setup of the physical cross-connect in the FastConnect location, confirm from your side that the light levels for each physical connection (cross-connect) are good (> -15 dBm). Don't proceed until they are.

In the Console, you can see the light levels that Oracle detects by viewing the details of the cross-connect and clicking **Light Levels**, as shown in the following screenshot:



If they are not good, contact your third-party network provider.

Task 6: Confirm your interfaces are up

Confirm your side of the interfaces for each physical connection (cross-connect) are up. Don't proceed until they are.

In the Console, you can see the status of Oracle's side of the interfaces (up or down) by viewing the details of the cross-connect and clicking **Light Levels**.

If the interfaces are not up, contact your third-party network provider.

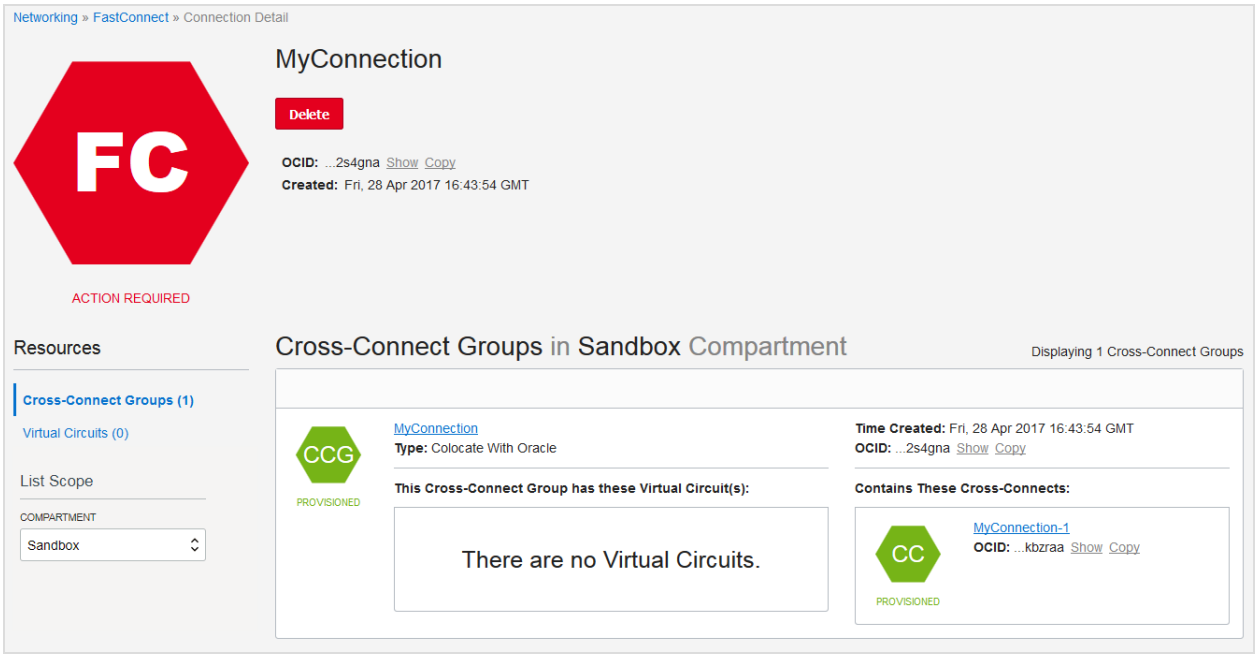
Task 7: Activate the cross-connect(s)

Summary: When your physical connection(s) in the FastConnect location are set up and ready to use, return to the Oracle Console and activate the cross-connect(s) that you set up earlier. This task informs Oracle that your physical network connection is ready. Oracle will then complete the router configuration for each cross-connect.

Instructions:

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **FastConnect**.
2. Select the compartment where the connection resides, and then click the connection to view its details. The FC icon is ACTION REQUIRED.
3. Click the cross-connect to view its details, and then click **Activate**.
4. Confirm when prompted.

If you have other cross-connects in this group that are ready to use, wait for the first to be provisioned, and then activate the next one. Only one cross-connect can be activated and then provisioned in a group at a time.



Summary of status icons:

- **FastConnect (FC) icon:** The FC status remains as ACTION REQUIRED to indicate that you have another action to take (see the next task).
- **Cross-connect group (CCG) icon:** The CCG status switches to PROVISIONED to indicate that the cross-connect group is ready to use.
- **Cross-connect (CC) icon:** The CC status switches to PROVISIONING and then changes to PROVISIONED (typically within one minute).

Task 8: Set up your virtual circuit(s)

Summary: Create one or more virtual circuits for your cross-connect group in the Oracle Console. The cross-connect group must be in the PROVISIONED state.

Instructions:

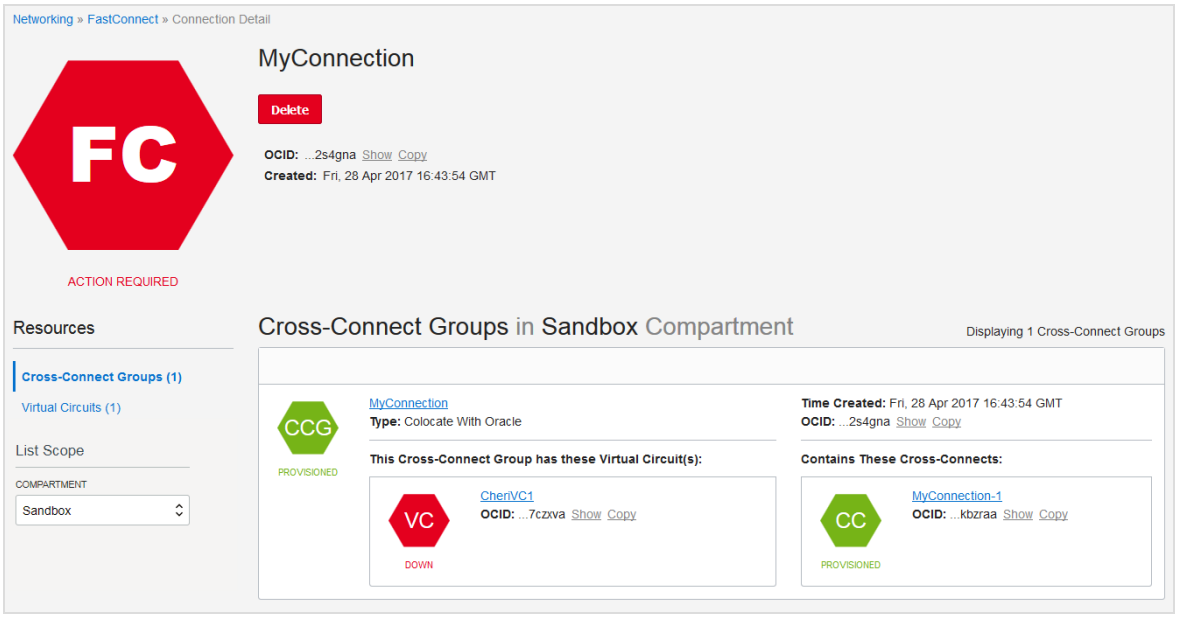
1. In the Console, return to the connection you created earlier, and click the **Virtual Circuits** tab on the left side of the page.
2. Click **Create Virtual Circuit**.
   In the resulting dialog box, you can add one or more virtual circuits to run on the cross-connect group.
3. Enter the following for your virtual circuit:
   - **Name:** A friendly name that helps you keep track of your virtual circuits. The value does not need to be unique across your virtual circuits, and you can change it later. Avoid entering confidential information.
   - **Create in Compartment:** Select the compartment where you want to create the virtual circuit. If you're not sure, select the current compartment (where the DRG resides). This choice of compartment, in conjunction with a corresponding IAM policy, controls who has access to the virtual circuit.
4. Choose the virtual circuit type (private or public). A private virtual circuit is for private peering (where your existing network receives your VCN's private IP addresses). A public virtual circuit is for public peering (where your existing network receives the Oracle Cloud Infrastructure regional public IP addresses). Also see Uses for FastConnect.
   - For a private virtual circuit, enter the following:
     - **Dynamic Routing Gateway Compartment:** Select the compartment where the DRG resides (it should already be selected).
     - **Dynamic Routing Gateway:** Select the DRG to route the FastConnect traffic to.
     - **Provisioned Bandwidth:** Choose your desired value. If your bandwidth needs increase later, you can update the virtual circuit to use a different value (see To edit a virtual circuit).
     - **Customer BGP ASN:** The public or private ASN for your network.

- **VLAN:** The number of the VLAN to use for this virtual circuit. It must be a VLAN that is not already assigned to another virtual circuit.
- **Customer BGP IP Address:** The BGP peering IP address for your edge, with either a /30 or /31 subnet mask.
- **Oracle BGP IP Address:** The BGP peering IP address you want to use for the Oracle edge, with either a /30 or /31 subnet mask.

- For a public virtual circuit, enter the following:
  - **Provisioned Bandwidth:** Choose your desired value. If your bandwidth needs increase later, you can update the virtual circuit to use a different value (see To edit a virtual circuit).
  - **Customer BGP ASN:** The public ASN for your network. Note that Oracle specifies the BGP IP addresses for a public virtual circuit.
  - **VLAN:** The number of the VLAN to use for this virtual circuit. It must be a VLAN that is not already assigned to another virtual circuit.
  - **Public IP Prefixes:** The public IP prefixes that you want Oracle to receive over the connection (each one must be /31 or less specific). You can enter a comma-separated list of prefixes, or one per line.

5. Click **Create**.

The virtual circuit is created. Its status is now included on the main connection's details.



Summary of status icons:

- **FastConnect (FC) icon:** The FC status switches to PROVISIONING briefly while Oracle's system provisions the virtual circuit. The status then switches to ACTION REQUIRED if the BGP session between your edge router and DRG is not yet correctly configured (see the next task), if the VLAN isn't configured correctly, or if there any other problems.
- **Cross-connect group (CCG) icon:** The CCG status remains as PROVISIONED.
- **Cross-connect (CC) icon:** The CC status remains as PROVISIONED.
- **Virtual circuit (VC) icon:** The virtual circuit's status is PROVISIONING briefly while Oracle's system provisions the virtual circuit. The status then switches to DOWN if the BGP session between your edge and Oracle's edge is not yet correctly configured, if the VLAN isn't configured correctly, or if there any other problems. Otherwise the status switches to UP.

Task 9: Configure your edge

Configure your edge router(s) to use the BGP information and VLAN for the virtual circuit. Oracle's BGP ASN is 31898. By default, Oracle uses the default BGP timers of 60 seconds for keep-alive and 180 seconds for hold-time. If you need fast BGP convergence, you can use any value in these supported ranges: 6-60 seconds for keep-alive, and 18-180 seconds for hold-time.



Important

If this is a public virtual circuit, Oracle's public IP addresses are advertised over both FastConnect and your connection with your Internet Service Provider (ISP). Make sure to give higher preference to FastConnect over your ISP.

Regarding LACP:

- LACP is required on the network interface that is directly plugged in to Oracle's router.
- LACP is required even if you have only a single cross-connect.
- If the third-party provider is performing any media conversion, LACP must be configured on the provider's device instead of your device.

Also configure the router for redundancy according to the network design you decided on earlier. After you successfully configure BGP and the VLAN, the virtual circuit's status will switch to UP.

Summary of status icons:

- **FastConnect (FC) icon:** The FC status switches to PROVISIONED when the BGP session is established. For a public virtual circuit, instead of switching to PROVISIONED, the status may switch to either IP CHECK IN PROGRESS or IP CHECK FAILED (if one of your public prefixes failed Oracle's verification). When Oracle successfully verifies all

the prefixes, the FC status switches to PROVISIONED.

- **Cross-connect group (CCG) icon:** The CCG status remains as PROVISIONED.
- **Cross-connect (CC) icon:**The CC status remains as PROVISIONED.
- **Virtual circuit (VC) icon:** The virtual circuit's status switches to UP when the BGP session is established.

Task 10: Ping the Oracle BGP IP address

Ping the Oracle BGP IP address assigned to the virtual circuit. Check the error counters and look for any dropped packets. Don't proceed until you can successfully ping this IP address without errors.

If the ping is not successful, and you're NOT learning MAC addresses, verify LACP is configured as mentioned in Task 9.

Task 11: Confirm the BGP session is established

For each virtual circuit you set up, confirm the BGP session is in an established state on your side of the connection.

Task 12: Test the connection

For a private virtual circuit: You should be able to launch an instance in your VCN and access it (for example, with SSH) from a host in your existing private network. See Creating an Instance. If you can, your FastConnect private virtual circuit is ready to use.

For a public virtual circuit:

1. Make sure that Oracle has successfully verified *at least one* of the public prefixes you've submitted. You can see the status of each prefix by viewing the virtual circuit's details in the Console. When one of the prefixes has been validated, Oracle starts advertising the regional OCI public addresses over the connection.
2. Launch an instance with a public IP address.

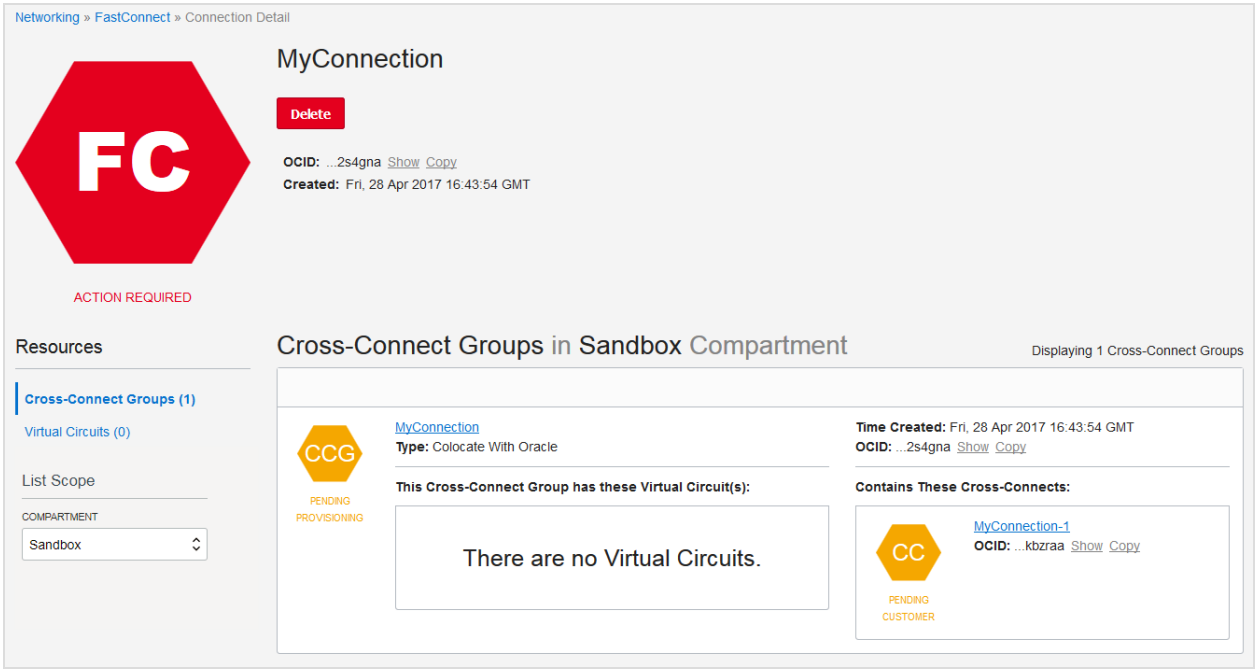
3. Ping the public IP address from a host in your existing private network. You should see the packet on the FastConnect interface on the virtual circuit. If you do, your FastConnect public virtual circuit is ready to use. However, remember that *only the public prefixes that Oracle has successfully verified so far* are advertised on the connection.

Managing Your Connection

To get the status of your connection

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **FastConnect**.
2. Click the connection you're interested in to view its details.

The following screenshot shows an example of the connection details, after you create the cross-connect group with a single cross-connect:



Here are reasons for particular status values for the overall connection:

ACTION REQUIRED:

- You need to request cabling at the FastConnect location for the cross-connect group you just created.
- Or, you need to activate a cross-connect (make sure it's ready to use first).
- Or, you need to set up at least one virtual circuit on your cross-connect group to complete setup for FastConnect.

DOWN:

In general this means you've created a virtual circuit, but configuration is incomplete or incorrect:

- You need to configure your edge.
- Or, you've configured BGP or the VLAN incorrectly on your edge (make sure to configure the router to use the BGP and VLAN values assigned to the virtual circuit).

IP CHECK IN PROGRESS:

For public virtual circuits only. This status means Oracle is in the process of verifying the public prefixes you've submitted. This status is shown if verification of at least one prefix is still in progress, and no prefixes have failed verification. You can get the status of each individual prefix you submitted by viewing the virtual circuit's details.

IP CHECK FAILED:

For public virtual circuits only. This means at least one of the public prefixes you've submitted failed Oracle's verification. That means Oracle will not advertise that prefix over the virtual circuit.

The following table summarizes the different states of each component involved in the connection at different points during setup:

Task in Setup Process	FC Icon (overall connection)	CCG Icon	CC Icon	VC Icon
Task 3: Set up your cross-connect group and cross-connect(s)	ACTION REQUIRED	PENDING PROVISIONING	PENDING CUSTOMER	N/A
Task 7: Activate the cross-connect (s)	ACTION REQUIRED	PROVISIONED	PROVISIONED	N/A

Task in Setup Process	FC Icon (overall connection)	CCG Icon	CC Icon	VC Icon
Task 8: Set up your virtual circuit(s)	PROVISIONING > PROVISIONED or DOWN	PROVISIONED	PROVISIONED	PROVISIONING > DOWN
Task 9: Configure your edge	DOWN > PROVISIONED For public virtual circuits, other possible values besides PROVISIONED: • ACTION REQUIRED (if you haven't yet submitted any prefixes to Oracle) • IP CHECK IN PROGRESS (if at least one prefix is still being validated) • IP CHECK FAILED (if at least one prefix failed)	PROVISIONED	PROVISIONED	DOWN > UP

To add a new cross-connect to an existing cross-connect group

When you first create a cross-connect group in the Console, you're allowed to create three cross-connects in the group. You can later add more to increase the bandwidth and resiliency of the group. The total allowed number is eight.

1. Create the new cross-connect in the existing cross-connect group:
    a. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **FastConnect**.
    b. Select the compartment where the connection resides, and then click the connection to view its details.
    c. Click the cross-connect group to view its details.
    d. Click **Add Cross-Connect to Group**.
    e. For the **Name**, enter a friendly name that helps you keep track of this cross-connect. The value does not need to be unique across your cross-connects. You cannot change this later in the Console. However, you can change it if you're using the API.
    f. Click **Create**.
    The cross-connect is created. The resulting dialog box has a link to the Letter of Authorization (LOA). You can get to the LOA again later by viewing the details of the cross-connect.
    g. Click the LOA link and print the LOA. You will need to submit it with your cabling order in the next step.
    h. Click **Close**.
    The overall status of the connection changes to ACTION REQUIRED to indicate that you have more work to do.
2. Perform tasks 4-7 in Getting Started with FastConnect. In summary, you need to have the cabling set up for the new cross-connect, validate the light levels and interfaces are good, and then activate the cross-connect.

After you activate the cross-connect, the status of the overall connection will be PROVISIONING until Oracle's system provisions the new cross-connect. Then the status will switch to PROVISIONED.

To edit a virtual circuit

You can change these items for a virtual circuit:

- The name
- The bandwidth
- Which DRG it uses (for a private virtual circuit)
- Which VLAN it uses
- The BGP session information
- The public IP prefixes (for a public virtual circuit)



Important

Notes About Editing a Virtual Circuit

If your virtual circuit is working and in the PROVISIONED state before you edit it, be aware that changing any of the properties besides the name, bandwidth, and public prefixes (for a public virtual circuit) causes the virtual circuit's state to switch to PROVISIONING and **may cause the related BGP session to go down.** After Oracle re-provisions the virtual circuit, its state returns to PROVISIONED. Make sure you confirm that the associated BGP session is back up.

If you change the public IP prefixes for a public virtual circuit, the BGP session and virtual circuit status is unaffected. However, the overall FastConnect status may change to IP CHECK IN PROGRESS or IP CHECK FAILED. Oracle begins advertising a new IP prefix over the connection only after verifying your ownership of it.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **FastConnect**.
2. Select the compartment where the connection resides, and then click the connection to view its details.
3. Click **Virtual Circuits**, and then click the virtual circuit to view its details.
4. Click **Edit** and make your changes.
5. Click **Save**.

To terminate a connection, or part of it

To stop being billed for a connection, you must terminate the virtual circuit, cross-connect(s), and cross-connect group associated with the connection (in that order).



Important

Also terminate the connection with the data center or third-party provider, or else they may continue to bill you.

To terminate a virtual circuit

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **FastConnect**.
2. Select the compartment where the connection resides, and then click the connection to view its details.
3. Click **Virtual Circuits**, and then click the virtual circuit to view its details.
4. Click **Delete**.
5. Confirm when prompted.

The virtual circuit's status changes to TERMINATING and then to TERMINATED.

To terminate a cross-connect

If you have multiple cross-connects to delete in a cross-connect group, wait until the state of the first one changes to TERMINATED before deleting the next one. Also, you can't delete a cross-connect if it's the *last* provisioned cross-connect in a cross-connect group that's being used by a provisioned virtual circuit.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **FastConnect**.
2. Select the compartment where the connection resides, and then click the connection to view its details.
3. Click the cross-connect you want to delete.
4. Click **Delete**.
5. Confirm when prompted.

The cross-connect's status changes to TERMINATING and then to TERMINATED.

To terminate a cross-connect group

Prerequisite: The cross-connect group must have no virtual circuits running on it and contain no cross-connects.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **FastConnect**.
2. Select the compartment where the connection resides, and then click the connection to view its details.
3. Click **Cross-Connect Groups**, and then click the cross-connect group to view its details.
4. Click **Delete**.
5. Confirm when prompted.

The cross-connect group's status changes to TERMINATING and then to TERMINATED.

To manage public IP prefixes for a public virtual circuit

For general information about the prefixes, see Logical Connection: Public Virtual Circuit.

You can specify your public IP prefixes when you create the virtual circuit. See Task 8: Set up your virtual circuit(s).

You can add or remove public IP prefixes later after creating the virtual circuit. See To edit a virtual circuit. If you add a new prefix, Oracle first verifies your company's ownership before advertising it across the connection. If you remove a prefix, Oracle stops advertising the prefix within a few minutes of your editing the virtual circuit.

You can view the state of Oracle's verification of a given public prefix by viewing the virtual circuit's details in the Console. Here are the possible values:

- IP CHECK IN PROGRESS: Oracle is in the process of verifying your organization's ownership of the prefix.
- IP CHECK FAILED: Oracle could not verify your organization's ownership. Oracle will not advertise the prefix over the virtual circuit.
- IP CHECK COMPLETED: Oracle successfully verified your organization's ownership. Oracle is advertising the prefix over the virtual circuit.

FastConnect: Colocation with Oracle

This topic is for customers who are colocated with Oracle in a FastConnect location. For a summary of the different ways to connect, see the connectivity models.

If you instead have a relationship with an Oracle provider, see FastConnect: With an Oracle Provider. Or if you have a relationship with a third-party provider, see FastConnect: With a Third-Party Provider.

For general information about FastConnect, see FastConnect.

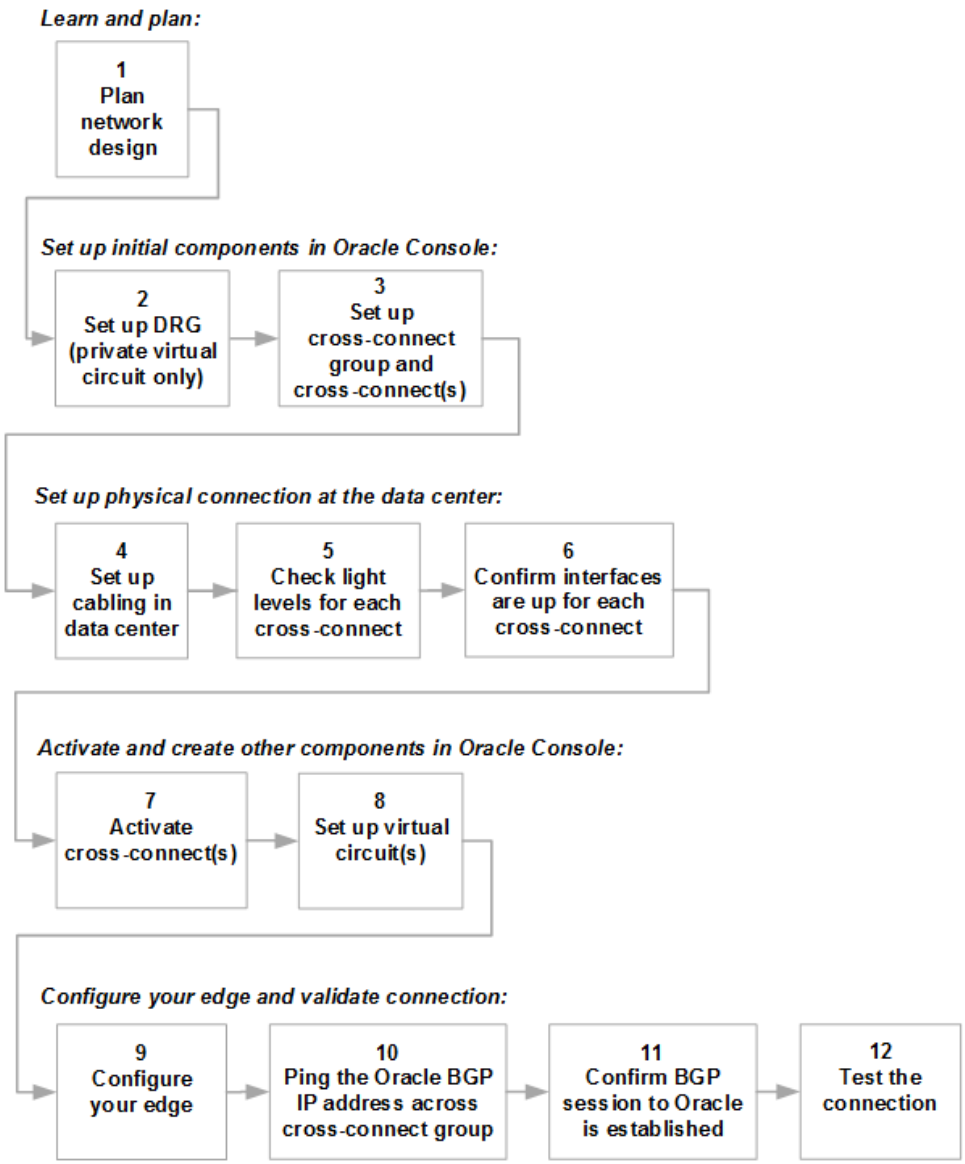


Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Getting Started with FastConnect

The following flow chart shows the overall process of setting up FastConnect.



Task 1: Learn and plan

If you haven't yet, walk through the planning in Before Getting Started: Learn and Plan.

Task 2: Set up a DRG (private peering only)

Summary: If you plan to use a private virtual circuit, you need a DRG. If you haven't already, use the Oracle Cloud Infrastructure Console to set up a DRG, attach it to your VCN, and update routing in your VCN to include a route rule to send traffic to the DRG. It's easy to forget to update the route table. Without the route rule, no traffic will flow.

Instructions:

- To create a dynamic routing gateway
- To attach a dynamic routing gateway to a VCN
- To update rules in an existing route table

Task 3: Set up your cross-connect group and cross-connect(s)

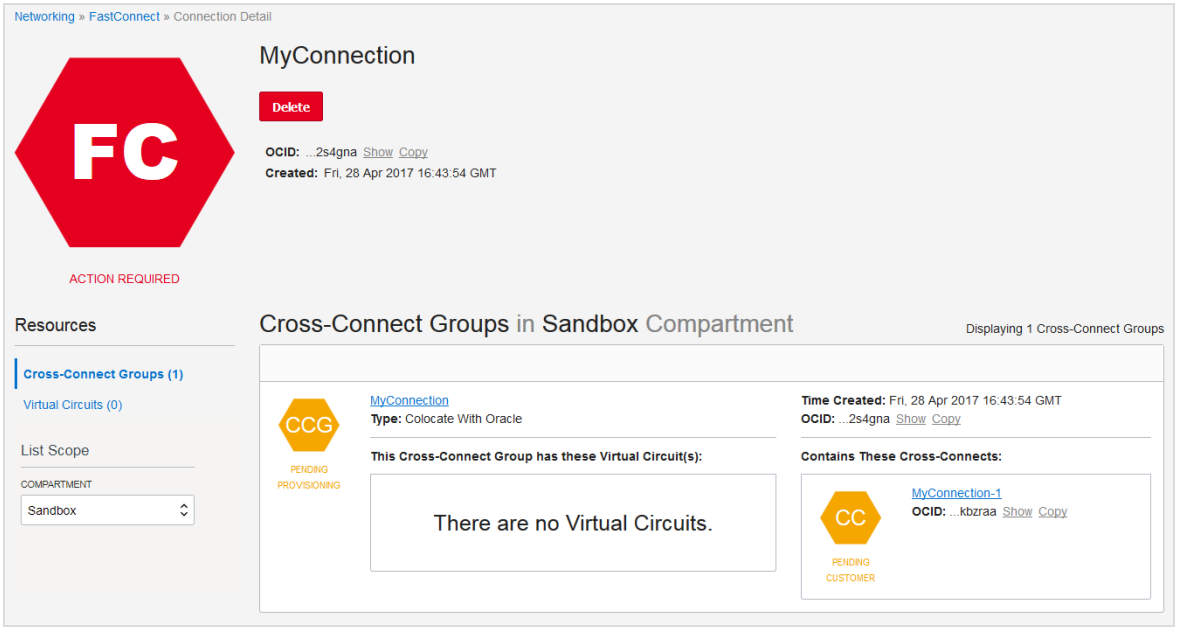
Summary: Create a connection in the Console, which consists of a cross-connect group that contains at least one cross-connect.

Instructions:

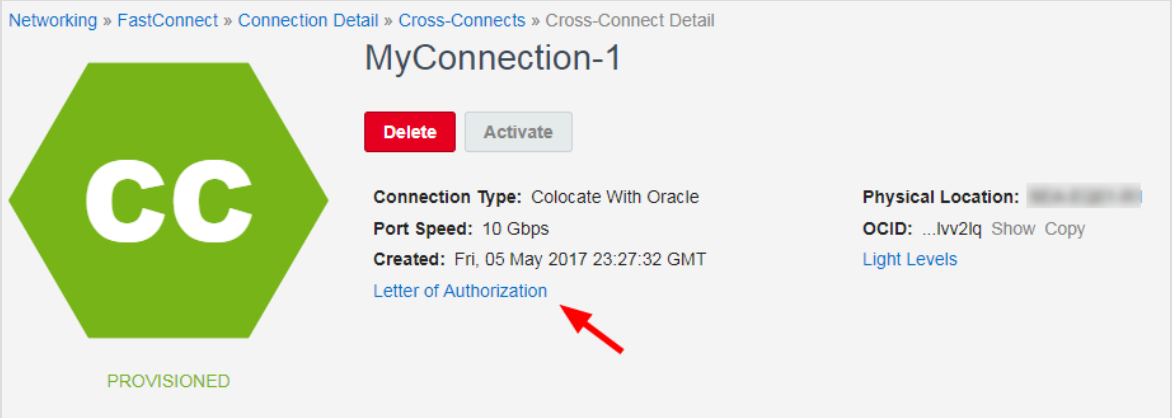
1. In the Console, confirm you're viewing the compartment that you want to work in. If you're not sure which one, use the compartment that contains the DRG that you'll connect to (for a private virtual circuit). This choice of compartment, in conjunction with a corresponding IAM policy, controls who has access to the cross-connect group and cross-connect(s) you're about to create.
2. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **FastConnect**.
   The resulting **FastConnect** page is where you'll create a new connection and later return to when you need to manage the connection and its components.

3. Click **Create Connection**.
4. Select the radio button for **Colocate with Oracle** and click **Continue**.
   In the resulting dialog box, you'll create a cross-connect group with up to three cross-connects in it. If you need more cross-connects in the group, you can add them later. You can have a maximum of eight cross-connects in a group.
5. Enter the following:
   - **Name:** A friendly name that helps you keep track of this connection. The cross-connect group will use this name. Each cross-connect in this group will also use this name, but with a hyphen and number appended (for example, MyName-1, MyName-2, and so on). You can't change the name later. Avoid entering confidential information.
   - **Create in Compartment:** Leave as is (the compartment you're currently working in).
   - **Number of cross-connects:** Select the number of individual cross-connects to create in the cross-connect group. In the Console, you can create three. If you need more, you can add more cross-connects later (total eight in a cross-connect group).
   - **Port speed:** All cross-connects must use a 10 Gbps port speed.
   - **Physical location:** Select the FastConnect location for this cross-connect group.
   - **Cross-Connect Group Proximity**: Appears only if you have one or more existing cross-connect groups in this FastConnect location. Here you may optionally specify whether you want the new cross-connect group to be on the same or different router than one of your other cross-connect groups.

6. Click **Continue**.
   The new connection is created and listed on the FastConnect page. Click it to see the connection details, which includes the status of each of the components (the cross-connect group, cross-connect(s), and later the virtual circuit(s)). The following screenshot shows the connection details:



7. Click each cross-connect to view its details (MyConnection-1 in the preceding screenshot), and then view and print the cross-connect's Letter of Authorization (LOA). You need to submit it with your cabling request at the FastConnect location in the next step.



Summary of status icons:

- **FastConnect (FC) icon:** The large icon in the top-left corner. It shows the general status of the overall FastConnect connection and whether you need to take action. At this point, the FC status will be ACTION REQUIRED (see the next task).
- **Cross-connect group (CCG) icon:** The icon near the middle of the page. It shows the status of the cross-connect group itself. At this point, the CCG status will be PENDING PROVISIONING.
- **Cross-connect (CC) icon:** The icon on the right side of the page. It shows the status of a given cross-connect. At this point, the CC status will be PENDING CUSTOMER.

Later, when you add a virtual circuit to your provisioned cross-connect group, under the CCG icon there will be a VC icon that shows the status of the virtual circuit.

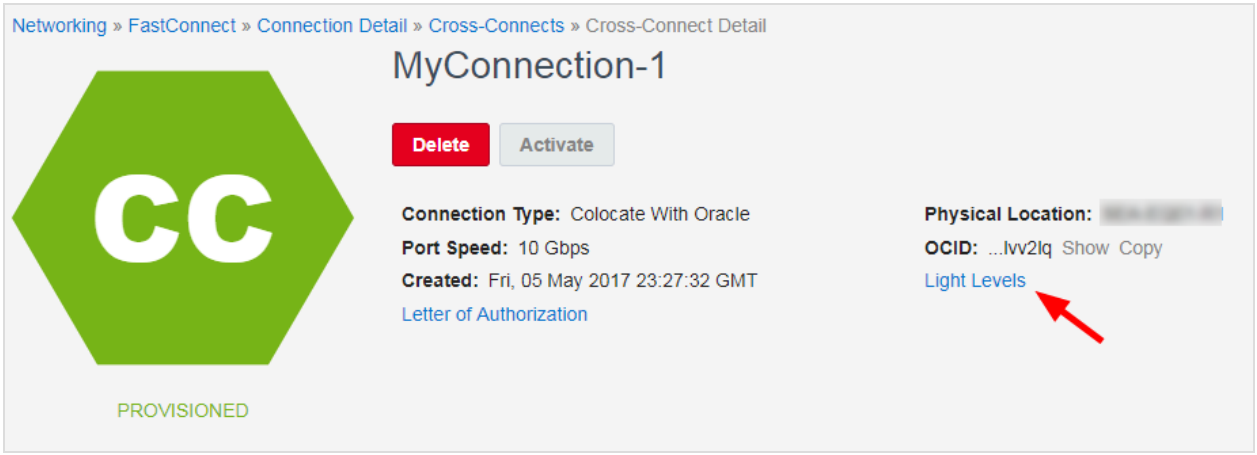
Task 4: Set up cabling in the FastConnect location

Submit the LOA(s) from the preceding task and request cabling at the FastConnect location. Each LOA is valid for a limited time. The details are printed on the LOA.

Task 5: Check light levels

Confirm from your side that the light levels for each physical connection (cross-connect) are good (> -15 dBm). Don't proceed until they are.

In the Console, you can see the light levels that Oracle detects by viewing the details of the cross-connect and clicking **Light Levels**, as shown in the following screenshot:



Task 6: Confirm your interfaces are up

Confirm your side of the interfaces for each physical connection (cross-connect) are up. Don't proceed until they are.

In the Console, you can see the status of Oracle's side of the interfaces (up or down) by viewing the details of the cross-connect and clicking **Light Levels**.

Task 7: Activate the cross-connect(s)

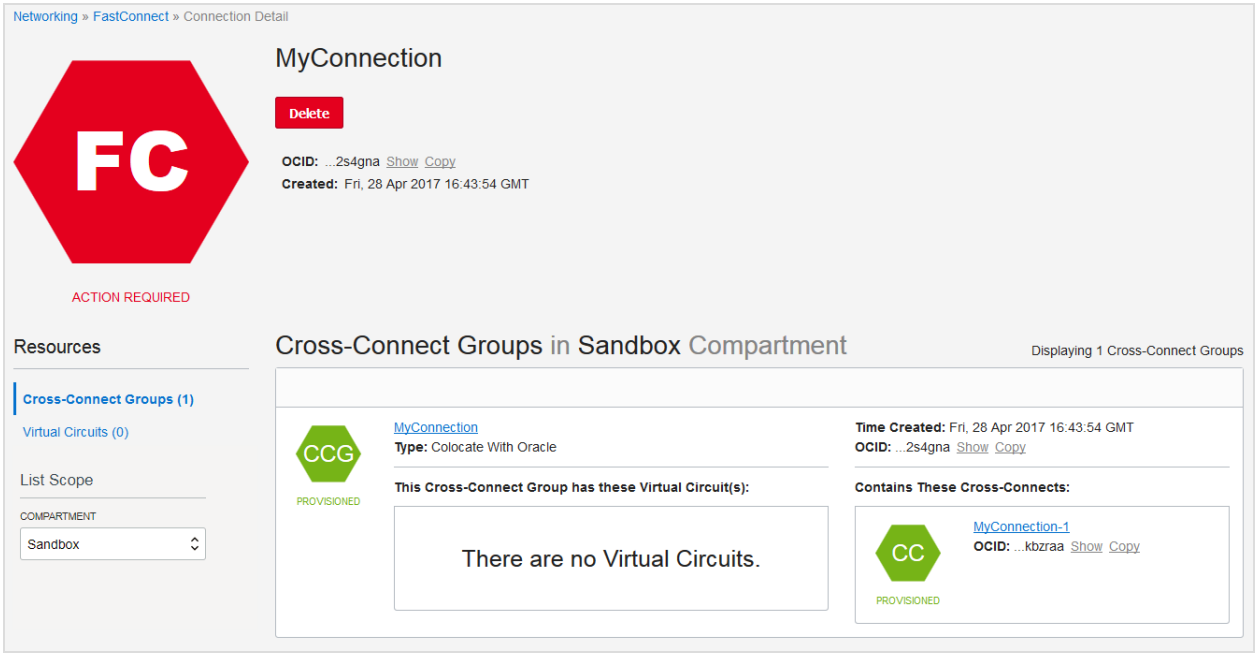
Summary: When your physical connection(s) in the FastConnect location are set up and ready to use, return to the Oracle Console and activate the cross-connect(s) that you set up earlier. This task informs Oracle that your physical network connection is ready. Oracle will

then complete the router configuration for each cross-connect.

Instructions:

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **FastConnect**.
2. Select the compartment where the connection resides, and then click the connection to view its details. The FC icon is ACTION REQUIRED.
3. Click the cross-connect to view its details, and then click **Activate**.
4. Confirm when prompted.

If you have other cross-connects in this group that are ready to use, wait for the first to be provisioned, and then activate the next one. Only one cross-connect can be activated and then provisioned in a group at a time.



Summary of status icons:

- **FastConnect (FC) icon:** The FC status remains as ACTION REQUIRED to indicate that you have another action to take (see the next task).
- **Cross-connect group (CCG) icon:** The CCG status switches to PROVISIONED to indicate that the cross-connect group is ready to use.
- **Cross-connect (CC) icon:** The CC status switches to PROVISIONING and then changes to PROVISIONED (typically within one minute).

Task 8: Set up your virtual circuit(s)

Summary: Create one or more virtual circuits for your cross-connect group in the Oracle Console. The cross-connect group must be in the PROVISIONED state.

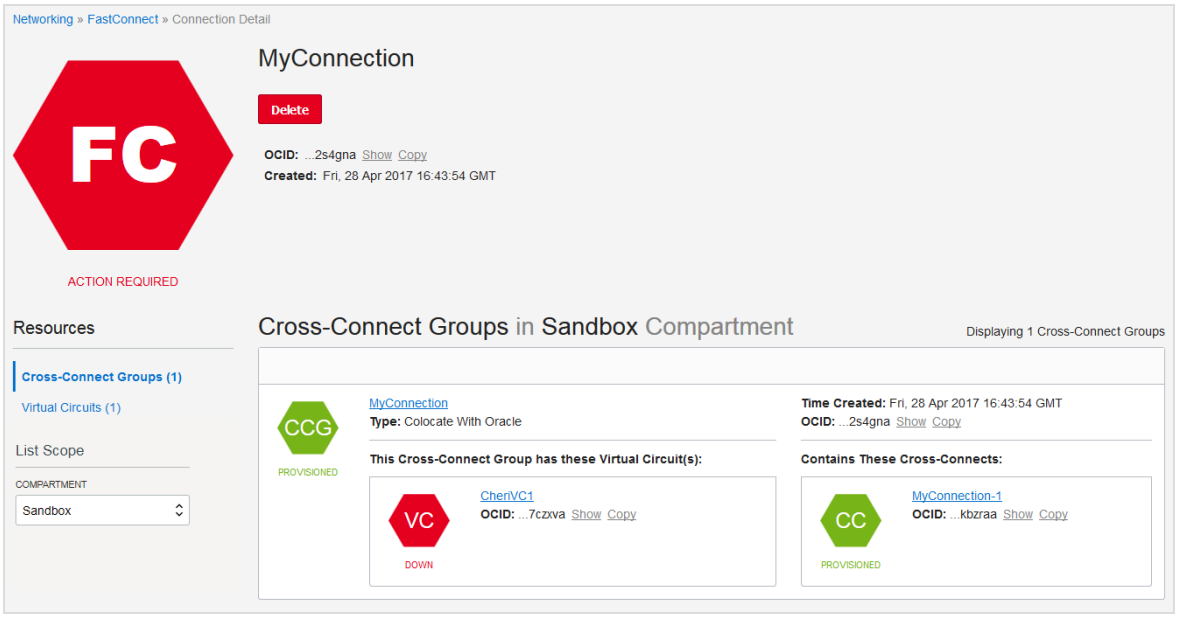
Instructions:

1. In the Console, return to the connection you created earlier, and click the **Virtual Circuits** tab on the left side of the page.
2. Click **Create Virtual Circuit**.
   In the resulting dialog box, you can add one or more virtual circuits to run on the cross-connect group.
3. Enter the following for your virtual circuit:
   - **Name:** A friendly name that helps you keep track of your virtual circuits. The value does not need to be unique across your virtual circuits, and you can change it later. Avoid entering confidential information.
   - **Create in Compartment:** Select the compartment where you want to create the virtual circuit. If you're not sure, select the current compartment (where the DRG resides). This choice of compartment, in conjunction with a corresponding IAM policy, controls who has access to the virtual circuit.
4. Choose the virtual circuit type (private or public). A private virtual circuit is for private peering (where your existing network receives your VCN's private IP addresses). A public virtual circuit is for public peering (where your existing network receives the

Oracle Cloud Infrastructure regional public IP addresses). Also see Uses for FastConnect.

- For a private virtual circuit, enter the following:
  - **Dynamic Routing Gateway Compartment:** Select the compartment where the DRG resides (it should already be selected).
  - **Dynamic Routing Gateway:** Select the DRG to route the FastConnect traffic to.
  - **Provisioned Bandwidth:** Choose your desired value. If your bandwidth needs increase later, you can update the virtual circuit to use a different value (see To edit a virtual circuit).
  - **Customer BGP ASN:** The public or private ASN for your network.
  - **VLAN:** The number of the VLAN to use for this virtual circuit. It must be a VLAN that is not already assigned to another virtual circuit.
  - **Customer BGP IP Address:** The BGP peering IP address for your edge, with either a /30 or /31 subnet mask.
  - **Oracle BGP IP Address:** The BGP peering IP address you want to use for the Oracle edge, with either a /30 or /31 subnet mask.
- For a public virtual circuit, enter the following:
  - **Provisioned Bandwidth:** Choose your desired value. If your bandwidth needs increase later, you can update the virtual circuit to use a different value (see To edit a virtual circuit).
  - **Customer BGP ASN:** The public ASN for your network. Note that Oracle specifies the BGP IP addresses for a public virtual circuit.
  - **VLAN:** The number of the VLAN to use for this virtual circuit. It must be a VLAN that is not already assigned to another virtual circuit.
  - **Public IP Prefixes:** The public IP prefixes that you want Oracle to receive over the connection (each one must be /31 or less specific). You can enter a comma-separated list of prefixes, or one per line.

5. Click **Create**.
   The virtual circuit is created. Its status is now included on the main connection's details.



Summary of status icons:

- **FastConnect (FC) icon:** The FC status switches to PROVISIONING briefly while Oracle's system provisions the virtual circuit. The status then switches to ACTION REQUIRED if the BGP session between your edge router and DRG is not yet correctly configured (see the next task), if the VLAN isn't configured correctly, or if there any other problems.
- **Cross-connect group (CCG) icon:** The CCG status remains as PROVISIONED.
- **Cross-connect (CC) icon:** The CC status remains as PROVISIONED.
- **Virtual circuit (VC) icon:** The virtual circuit's status is PROVISIONING briefly while Oracle's system provisions the virtual circuit. The status then switches to DOWN if the BGP session between your edge and Oracle's edge is not yet correctly configured, if the

VLAN isn't configured correctly, or if there any other problems. Otherwise the status switches to UP.

Task 9: Configure your edge

Configure your edge router(s) to use the BGP information and VLAN for the virtual circuit. Oracle's BGP ASN is 31898. By default, Oracle uses the default BGP timers of 60 seconds for keep-alive and 180 seconds for hold-time. If you need fast BGP convergence, you can use any value in these supported ranges: 6-60 seconds for keep-alive, and 18-180 seconds for hold-time.



Important

If this is a public virtual circuit, Oracle's public IP addresses are advertised over both FastConnect and your connection with your Internet Service Provider (ISP). Make sure to give higher preference to FastConnect over your ISP.

Regarding LACP:

- LACP is required on the network interface that is directly plugged in to Oracle's router.
- LACP is required even if you have only a single cross-connect.

Also configure the router for redundancy according to the network design you decided on earlier. After you successfully configure BGP and the VLAN, the virtual circuit's status will switch to UP.

Summary of status icons:

- **FastConnect (FC) icon:** The FC status switches to PROVISIONED when the BGP session is established. For a public virtual circuit, instead of switching to PROVISIONED, the status may switch to either IP CHECK IN PROGRESS or IP CHECK FAILED (if one of

your public prefixes failed Oracle's verification). When Oracle successfully verifies all the prefixes, the FC status switches to PROVISIONED.

- **Cross-connect group (CCG) icon:** The CCG status remains as PROVISIONED.
- **Cross-connect (CC) icon:**The CC status remains as PROVISIONED.
- **Virtual circuit (VC) icon:** The virtual circuit's status switches to UP when the BGP session is established.

Task 10: Ping the Oracle BGP IP address

Ping the Oracle BGP IP address assigned to the virtual circuit. Check the error counters and look for any dropped packets. Don't proceed until you can successfully ping this IP address without errors.

If the ping is not successful, and you're NOT learning MAC addresses, verify that you configured LACP as mentioned in Task 9.

Task 11: Confirm the BGP session is established

For each virtual circuit you set up, confirm the BGP session is in an established state on your side of the connection.

Task 12: Test the connection

For a private virtual circuit: You should be able to launch an instance in your VCN and access it (for example, with SSH) from a host in your existing private network. See Creating an Instance. If you can, your FastConnect private virtual circuit is ready to use.

For a public virtual circuit:

1. Make sure that Oracle has successfully verified *at least one* of the public prefixes you've submitted. You can see the status of each prefix by viewing the virtual circuit's details in the Console. When one of the prefixes has been validated, Oracle starts advertising the regional OCI public addresses over the connection.

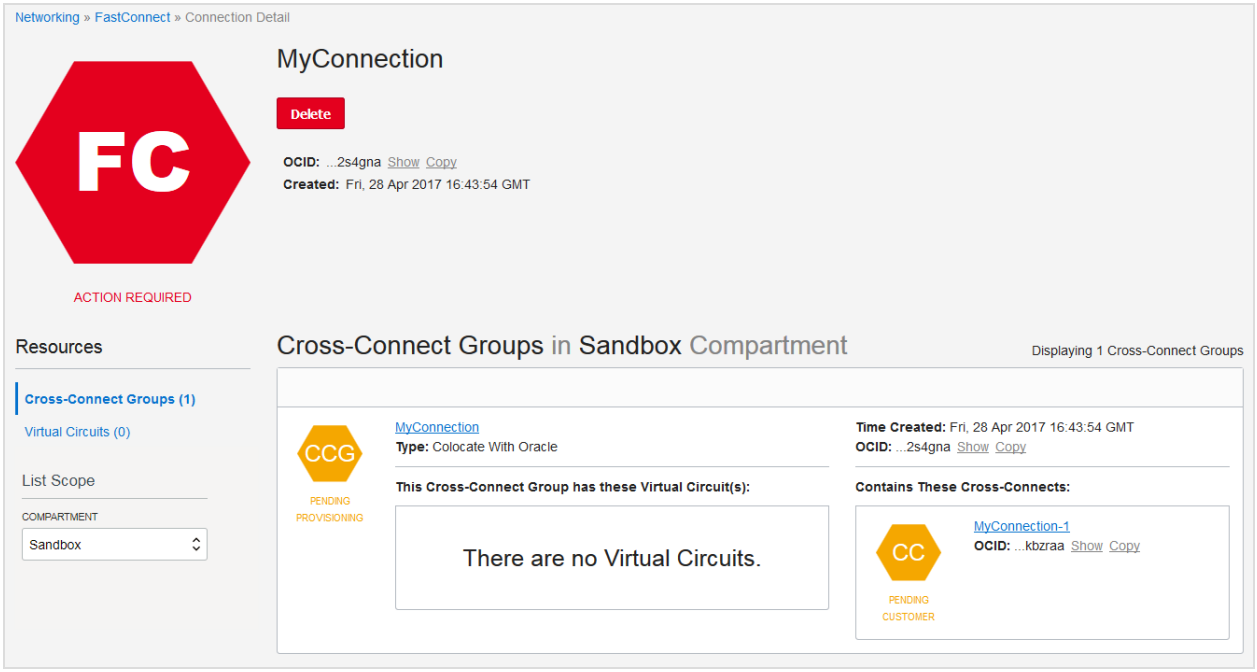
2. Launch an instance with a public IP address.
3. Ping the public IP address from a host in your existing private network. You should see the packet on the FastConnect interface on the virtual circuit. If you do, your FastConnect public virtual circuit is ready to use. However, remember that *only the public prefixes that Oracle has successfully verified so far* are advertised on the connection.

Managing Your Connection

To get the status of your connection

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **FastConnect**.
2. Click the connection you're interested in to view its details.

The following screenshot shows an example of the connection details, after you create the cross-connect group with a single cross-connect:



Here are reasons for particular status values for the overall connection:

ACTION REQUIRED:

- You need to request cabling at the FastConnect location for the cross-connect group you just created.
- Or, you need to activate a cross-connect (make sure it's ready to use first).
- Or, you need to set up at least one virtual circuit on your cross-connect group to complete setup for FastConnect.

DOWN:

In general this means you've created a virtual circuit, but configuration is incomplete or incorrect:

- You need to configure your edge.
- Or, you've configured BGP or the VLAN incorrectly on your edge (make sure to configure the router to use the BGP and VLAN values assigned to the virtual circuit).

IP CHECK IN PROGRESS:

For public virtual circuits only. This status means Oracle is in the process of verifying the public prefixes you've submitted. This status is shown if verification of at least one prefix is still in progress, and no prefixes have failed verification. You can get the status of each individual prefix you submitted by viewing the virtual circuit's details.

IP CHECK FAILED:

For public virtual circuits only. This means at least one of the public prefixes you've submitted failed Oracle's verification. That means Oracle will not advertise that prefix over the virtual circuit.

The following table summarizes the different states of each component involved in the connection at different points during setup:

Task in Setup Process	FC Icon (overall connection)	CCG Icon	CC Icon	VC Icon
Task 3: Set up your cross-connect group and cross-connect(s)	ACTION REQUIRED	PENDING PROVISIONING	PENDING CUSTOMER	N/A
Task 7: Activate the cross-connect (s)	ACTION REQUIRED	PROVISIONED	PROVISIONED	N/A

Task in Setup Process	FC Icon (overall connection)	CCG Icon	CC Icon	VC Icon
Task 8: Set up your virtual circuit(s)	PROVISIONING > PROVISIONED or DOWN	PROVISIONED	PROVISIONED	PROVISIONING > DOWN
Task 9: Configure your edge	DOWN > PROVISIONED For public virtual circuits, other possible values besides PROVISIONED: • ACTION REQUIRED (if you haven't yet submitted any prefixes to Oracle) • IP CHECK IN PROGRESS (if at least one prefix is still being validated) • IP CHECK FAILED (if at least one prefix failed)	PROVISIONED	PROVISIONED	DOWN > UP

To add a new cross-connect to an existing cross-connect group

When you first create a cross-connect group in the Console, you're allowed to create three cross-connects in the group. You can later add more to increase the bandwidth and resiliency of the group. The total allowed number is eight.

1. Create the new cross-connect in the existing cross-connect group:
    a. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **FastConnect**.
    b. Select the compartment where the connection resides, and then click the connection to view its details.
    c. Click the cross-connect group to view its details.
    d. Click **Add Cross-Connect to Group**.
    e. For the **Name**, enter a friendly name that helps you keep track of this cross-connect. The value does not need to be unique across your cross-connects. You cannot change this later in the Console. However, you can change it if you're using the API.
    f. Click **Create**.
    The cross-connect is created. The resulting dialog box has a link to the Letter of Authorization (LOA). You can get to the LOA again later by viewing the details of the cross-connect.
    g. Click the LOA link and print the LOA. You will need to submit it with your cabling order in the next step.
    h. Click **Close**.
    The overall status of the connection changes to ACTION REQUIRED to indicate that you have more work to do.
2. Perform tasks 4-7 in Getting Started with FastConnect. In summary, you need to have the cabling set up for the new cross-connect, validate the light levels and interfaces are good, and then activate the cross-connect.

After you activate the cross-connect, the status of the overall connection will be PROVISIONING until Oracle's system provisions the new cross-connect. Then the status will switch to PROVISIONED.

To edit a virtual circuit

You can change these items for a virtual circuit:

- The name
- The bandwidth
- Which DRG it uses (for a private virtual circuit)
- Which VLAN it uses
- The BGP session information
- The public IP prefixes (for a public virtual circuit)

Important

Notes About Editing a Virtual Circuit

If your virtual circuit is working and in the PROVISIONED state before you edit it, be aware that changing any of the properties besides the name, bandwidth, and public prefixes (for a public virtual circuit) causes the virtual circuit's state to switch to PROVISIONING and **may cause the related BGP session to go down.** After Oracle re-provisions the virtual circuit, its state returns to PROVISIONED. Make sure you confirm that the associated BGP session is back up.

If you change the public IP prefixes for a public virtual circuit, the BGP session and virtual circuit status is unaffected. However, the overall FastConnect status may change to IP CHECK IN PROGRESS or IP CHECK FAILED. Oracle begins advertising a new IP prefix over the connection only after verifying your ownership of it.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **FastConnect**.
2. Select the compartment where the connection resides, and then click the connection to view its details.
3. Click **Virtual Circuits**, and then click the virtual circuit to view its details.
4. Click **Edit** and make your changes.
5. Click **Save**.

To terminate a connection, or part of it

To stop being billed for a connection, you must terminate the virtual circuit, cross-connect(s), and cross-connect group associated with the connection (in that order).

To terminate a virtual circuit

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **FastConnect**.
2. Select the compartment where the connection resides, and then click the connection to view its details.
3. Click **Virtual Circuits**, and then click the virtual circuit to view its details.
4. Click **Delete**.
5. Confirm when prompted.

The virtual circuit's status changes to TERMINATING and then to TERMINATED.

To terminate a cross-connect

If you have multiple cross-connects to delete in a cross-connect group, wait until the state of the first one changes to TERMINATED before deleting the next one. Also, you can't delete a cross-connect if it's the *last* provisioned cross-connect in a cross-connect group that's being used by a provisioned virtual circuit.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **FastConnect**.
2. Select the compartment where the connection resides, and then click the connection to view its details.
3. Click the cross-connect you want to delete.

4. Click **Delete**.
5. Confirm when prompted.

The cross-connect's status changes to TERMINATING and then to TERMINATED.

To terminate a cross-connect group

Prerequisite: The cross-connect group must have no virtual circuits running on it and contain no cross-connects.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **FastConnect**.
2. Select the compartment where the connection resides, and then click the connection to view its details.
3. Click **Cross-Connect Groups**, and then click the cross-connect group to view its details.
4. Click **Delete**.
5. Confirm when prompted.

The cross-connect group's status changes to TERMINATING and then to TERMINATED.

To manage public IP prefixes for a public virtual circuit

For general information about the prefixes, see Logical Connection: Public Virtual Circuit.

You can specify your public IP prefixes when you create the virtual circuit. See Task 8: Set up your virtual circuit(s).

You can add or remove public IP prefixes later after creating the virtual circuit. See To edit a virtual circuit. If you add a new prefix, Oracle first verifies your company's ownership before advertising it across the connection. If you remove a prefix, Oracle stops advertising the prefix within a few minutes of your editing the virtual circuit.

You can view the state of Oracle's verification of a given public prefix by viewing the virtual circuit's details in the Console. Here are the possible values:

- IP CHECK IN PROGRESS: Oracle is in the process of verifying your organization's ownership of the prefix.
- IP CHECK FAILED: Oracle could not verify your organization's ownership. Oracle will not advertise the prefix over the virtual circuit.
- IP CHECK COMPLETED: Oracle successfully verified your organization's ownership. Oracle is advertising the prefix over the virtual circuit.

FastConnect Troubleshooting

This topic covers troubleshooting techniques for a FastConnect connection that has issues.

Some of the troubleshooting techniques assume that you are a network engineer with access to your CPE's configuration.

General Issues

FastConnect is DOWN



Important

If you're working with an Oracle provider or a third-party provider, contact both the provider and Oracle for help troubleshooting the issue. If you're colocated with Oracle, contact Oracle.

Cross-connect and physical connection (layer 1)

Check these items:

- **Port allocation:** Verify that your connection is using the correct port, and the port is UP and activated.
- **Optical signal:** Verify that your connection is using the correct optics and transceiver, and the port is sending and receiving an optimal signal. For more information, see FastConnect Requirements.
- **Fiber strands:** Try rolling or flipping the Tx/Rx fiber strands.
- **End-to-end physical connectivity:** Verify the end-to-end physical connectivity. Also verify the Tx/Rx optic signal between your CPE, the provider's network device (if you're working with a provider), and the Oracle FastConnect router.

Data-link (layer 2)

Check the following items on your CPE. If you're working with a provider, also have them check the items on their network device:

- **BGP address:** Verify that the router is configured with the correct BGP peering IP address under the correct VLAN on the interface.
- **MAC address:** Verify that the router is receiving the MAC address from the Oracle FastConnect router, and that the MAC address has an entry in the router's address resolution protocol (ARP) table.
- **LAG and LACP:** Verify that the router has LAG configured and LACP is enabled on the interface (the Oracle FastConnect router requires both). For more information, see FastConnect Requirements.

Network and transport (layers 3 and 4)

Check the following items on your CPE. If you're working with a provider, also have them

check the items on their network device:

- **BGP address:** Verify that the router is configured with the correct BGP peering IP address.
- **ASN:** Verify that the router is configured with the correct BGP local ASN and Oracle ASN (31898).
- **MD5:** Verify that MD5 authentication is disabled or not configured on the router. Oracle FastConnect does not support MD5 authentication.
- **Maximum prefixes:** Verify that you are advertising no more than the maximum allowed number of prefixes for virtual circuits. If you're advertising more, BGP won't be established. Here are the limits:
  - Public virtual circuits: maximum 200 prefixes
  - Private virtual circuits: maximum 2000 prefixes
- **Firewalls:** Verify that your on-premises firewall or access control lists are not blocking TCP port 179 (BGP) or any high-numbered TCP ports.

FastConnect virtual circuit is UP, but no traffic is passing through

Check these items:

- **VCN security lists:** Ensure you've set up the VCN security lists to allow the desired traffic (both ingress and egress rules). Note that the VCN's default security list does not allow ping traffic (ICMP type 8 and ICMP type 0). You must add the appropriate ingress and egress rules to allow ping traffic.
- **Correct routes on both ends:** Verify that you have received the correct VCN routes from FastConnect and the CPE is using those routes. Likewise, verify that you are advertising the correct on-premises network routes to FastConnect and the VCN route tables use those routes.

FastConnect virtual circuit is UP, but traffic is passing in only one direction

Check these items:

- **VCN security lists:** Ensure that your VCN security lists allow traffic in both directions (ingress and egress).
- **Firewalls:** Verify that your on-premises firewall or access control lists are not blocking traffic to or from the Oracle end.
- **Asymmetric routing:** Oracle uses asymmetric routing. If you have multiple virtual circuits, make sure that your CPE is configured for asymmetric route processing.
- **Redundant connections:** If you have redundant FastConnect virtual circuits, make sure they're both advertising the same routes.

Redundant Connections

Keep in mind that FastConnect uses BGP dynamic routing. Oracle IPSec VPN connections use static routing and do not support BGP.

IPSec and FastConnect are both set up, but traffic is only passing through IPSec

If you use the same routes for both the IPSec VPN and FastConnect, Oracle prefers the oldest stable connection. Make sure to use more specific routes for the connection you want as primary.

Internet Gateway

This topic describes how to set up and manage an internet gateway to give your VCN internet access.



Tip

Oracle also offers a NAT gateway, which is recommended for subnets in your VCN that do not require ingress connections from the internet.

Highlights

- An internet gateway is an optional virtual router you can add to your VCN to enable direct connectivity to the internet.
- The gateway supports connections initiated from within the VCN (egress) and connections initiated from the internet (ingress).
- Resources that need to use the gateway for internet access must be in a public subnet and have public IP addresses. Resources that have private IP addresses can instead use a NAT gateway to initiate connections to the internet.
- Each public subnet that needs to use the internet gateway must have a route table rule that specifies the gateway as the target.
- You use each public subnet's security list rules to control the types of traffic allowed in and out of resources in that subnet. For a public subnet, make sure to allow only the desired types of internet traffic.
- The internet gateway can be used only by resources in the gateway's VCN. Hosts in the connected on-premises network or in a peered VCN cannot use that internet gateway.

Overview of Internet Gateways

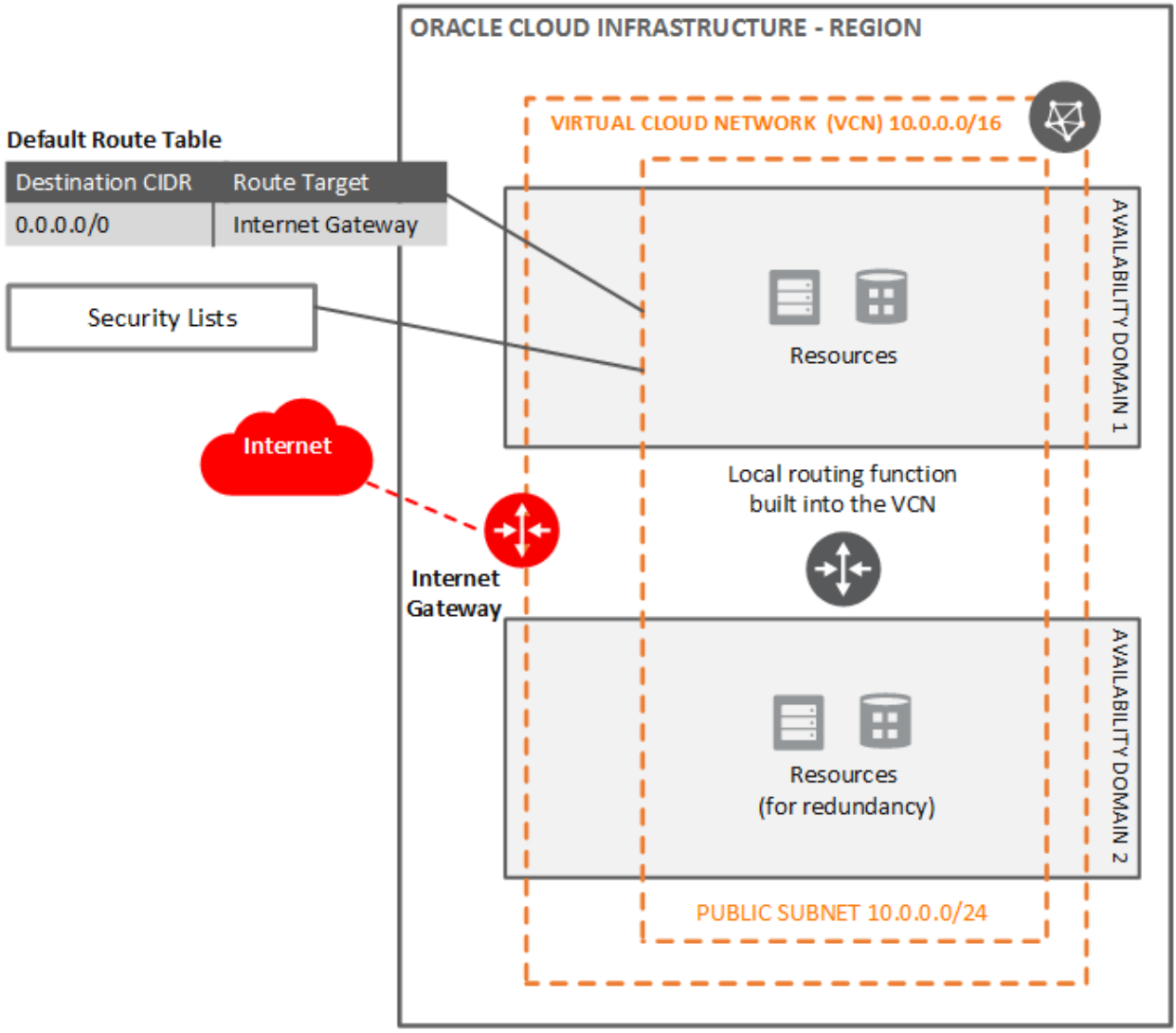
Before continuing, make sure you've read Access to the Internet and also understand how to set up security list rules for the resources in a subnet.

An internet gateway as an optional virtual router that connects the edge of the VCN with the internet. To use the gateway, the hosts on both ends of the connection must have public IP addresses for routing. Connections that originate in your VCN and are destined for a public IP

address (either inside or outside the VCN) go through the internet gateway. Connections that originate outside the VCN and are destined for a public IP address inside the VCN go through the internet gateway.

A given VCN can have only one internet gateway. You control which public subnets in the VCN can use the gateway by configuring the subnet's associated route table and security lists.

The following diagram illustrates a simple VCN setup with two public subnets. The VCN has an internet gateway, and the two public subnets are both configured to use the VCN's default route table. The table has a route rule that sends all egress traffic from the subnets to the internet gateway. The gateway allows any ingress connections from the internet with a destination IP address equal to the public IP address of a resource in the VCN. However, the public subnet's security list rules ultimately determine the specific *types* of traffic that are allowed in and out of the resources in the subnet. Those specific security list rules are not shown in the diagram.


ORACLE CLOUD INFRASTRUCTURE - REGION
VIRTUAL CLOUD NETWORK (VCN) 10.0.0.0/16
Default Route Table
Destination CIDR
Route Target
0.0.0.0/0
Internet Gateway
Security Lists
Internet
Internet Gateway
AVAILABILITY DOMAIN 1
Resources
Local routing function built into the VCN
AVAILABILITY DOMAIN 2
Resources (for redundancy)
PUBLIC SUBNET 10.0.0.0/24



Tip

When an internet gateway receives traffic from your VCN destined for a public IP address that is part of Oracle Cloud Infrastructure (such as Object Storage), the internet gateway routes the traffic to the destination without sending the traffic over the internet.

Working with Internet Gateways

You create an internet gateway in the context of a specific VCN. In other words, the internet gateway is automatically attached to a VCN. However, you can disable and re-enable the internet gateway at any time. Compare this with a dynamic routing gateway (DRG), which you create as a standalone object that you then *attach* to a particular VCN. DRGs use a different model because they're intended to be modular building blocks for privately connecting VCNs to your on-premises network.

For traffic to flow between a subnet and an internet gateway, you must create a route rule accordingly in the subnet's route table (for example, destination CIDR = 0.0.0.0/0 and target = internet gateway). If the internet gateway is disabled, that means no traffic will flow to or from the internet even if there's a route rule that enables that traffic. For more information, see Route Tables.

For the purposes of access control, you must specify the compartment where you want the internet gateway to reside. If you're not sure which compartment to use, put the internet gateway in the same compartment as the cloud network. For more information, see Access Control.

You may optionally assign a friendly name to the internet gateway. It doesn't have to be unique, and you can change it later. Oracle automatically assigns the internet gateway a unique identifier called an Oracle Cloud ID (OCID). For more information, see Resource Identifiers.

To delete an internet gateway, it does not have to be disabled, but there must not be a route table that lists it as a target.

Using the Console



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

To set up an internet gateway

Prerequisites:

- You've determined which subnets in the VCN need access to the internet, and you've created those public subnets.
- You've determined the types of ingress and egress internet traffic that you want to enable for the resources in each public subnet (examples: ingress HTTPS connections, ingress ICMP ping connections).
- The required IAM policy is in place to allow you to work with Networking service resources. For administrators: see IAM Policies for Networking.



Important

If you've configured the public subnet to use the default security list, remember that the list includes several helpful default rules that enable typical required access (examples: ingress SSH, egress access to all destinations). Oracle recommends that you use the default security list and understand the access it provides. If you choose not to use the list, make sure you understand the implications and enable basic required access in the public subnet's custom security list.

1. For each public subnet that needs to use the internet gateway, set up the subnet's security list rules to allow the desired internet traffic:
    a. In the Console, while viewing the VCN you're interested in, click **Security Lists**.
    b. Click the security list you're interested in (a security list associated with the public subnet).
    c. Under **Resources**, click either **Ingress Rules** or **Egress Rules** depending on the type of rule you want to work with.
    d. If you want to add a new rule, click **Add Ingress Rule** (or **Add Egress Rule**).

    ### Example

    Imagine you have web servers in the public subnet. This example shows how to add an ingress rule for HTTPS connections (TCP port 443) coming from the internet to the web server. Without this rule, inbound HTTPS connections are not allowed.

i. Leave the **Stateless** check box unselected.
ii. **Source Type:** CIDR
iii. **Source CIDR:** 0.0.0.0/0
iv. **IP Protocol:** Leave as TCP.
v. **Source Port Range:** Leave as All.
vi. **Destination Port Range**: Enter 443.

e. If you want to delete an existing rule, click the Actions icon (three dots), and then click **Remove**.
f. If you wanted to edit an existing rule, click the Actions icon (three dots), and then click **Edit**.

2. Create the VCN's internet gateway:
   a. In the Console, while viewing the VCN you're interested in, click **Internet Gateways**
   b. Click **Create Internet Gateway**.
   c. Enter the following:
      - **Name:** A friendly name for the internet gateway. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
      - **Create in Compartment:** The compartment where you want to create the internet gateway, if different from the compartment you're currently working in.
      - **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you

are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

d. Click **Create Internet Gateway**.

Your internet gateway is created and displayed on the **Internet Gateways** page of the compartment you chose. It's already enabled, but you still need to add a route rule that allows traffic to flow to the gateway.

3. For each public subnet that needs to use the internet gateway, update the subnet's route table:
   a. While viewing the VCN's details, click **Route Tables**.
   b. Click the public subnet's route table to view its details.
   c. Click **Add Route Rule**.
   d. Enter the following:
      - **Target Type:** Internet Gateway
      - **Destination CIDR block:** 0.0.0.0/0 (which means that all non-intra-VCN traffic that is not already covered by other rules in the route table will go to the target specified in this rule)
      - **Compartment:** The compartment where the internet gateway is located.
      - **Target:** The internet gateway you just created.
   e. Click **Save**.

An internet gateway is now enabled and working for your cloud network.

To disable/enable an internet gateway

This is available only through the API. If you don't have access to the API and need to disable or enable an internet gateway, contact Oracle Support. You can also easily delete and recreate the internet gateway if needed. Just make sure to update any route tables that refer to the internet gateway.

To delete an internet gateway

Prerequisite: The internet gateway does not have to be disabled, but there must not be a route table that lists it as a target.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Under **Resources**, click **Internet Gateways**.
4. Click the Actions icon (three dots) for the internet gateway, and then click **Terminate**.
5. Confirm when prompted.

To manage tags for an internet gateway

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Under **Resources**, click **Internet Gateways**.
4. Click the Actions icon (three dots) for the internet gateway, and then click **View Tags**. From there you can view the existing tags, edit them, and apply new ones.

For more information, see Resource Tags.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

To manage your internet gateways, use these operations:

- ListInternetGateways
- CreateInternetGateway

- GetInternetGatway
- UpdateInternetGateway
- DeleteInternetGateway

NAT Gateway

This topic describes how to set up and manage a Network Address Translation (NAT) gateway. A NAT gateway gives cloud resources without public IP addresses access to the internet without exposing those resources to incoming internet connections.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Highlights

- You can add a NAT gateway to your VCN to give instances in a private subnet access to the internet.
- Instances in a private subnet don't have public IP addresses. With the NAT gateway, they can initiate connections to the internet and receive responses, but not receive inbound connections initiated from the internet.
- NAT gateways are highly available and support TCP, UDP, and ICMP ping traffic.

Overview of NAT

NAT is a networking technique commonly used to give an entire private network access to the internet without assigning each host a public IPv4 address. The hosts can initiate connections

to the internet and receive responses, but not receive inbound connections initiated from the internet.

When a host in the private network initiates an internet-bound connection, the NAT device's public IP address becomes the source IP address for the outbound traffic. The response traffic from the internet therefore uses that public IP address as the destination IP address. The NAT device then routes the response to the host in the private network that initiated the connection.

Overview of NAT Gateways

The Networking service offers a reliable and highly available NAT solution for your VCN in the form of a NAT gateway.

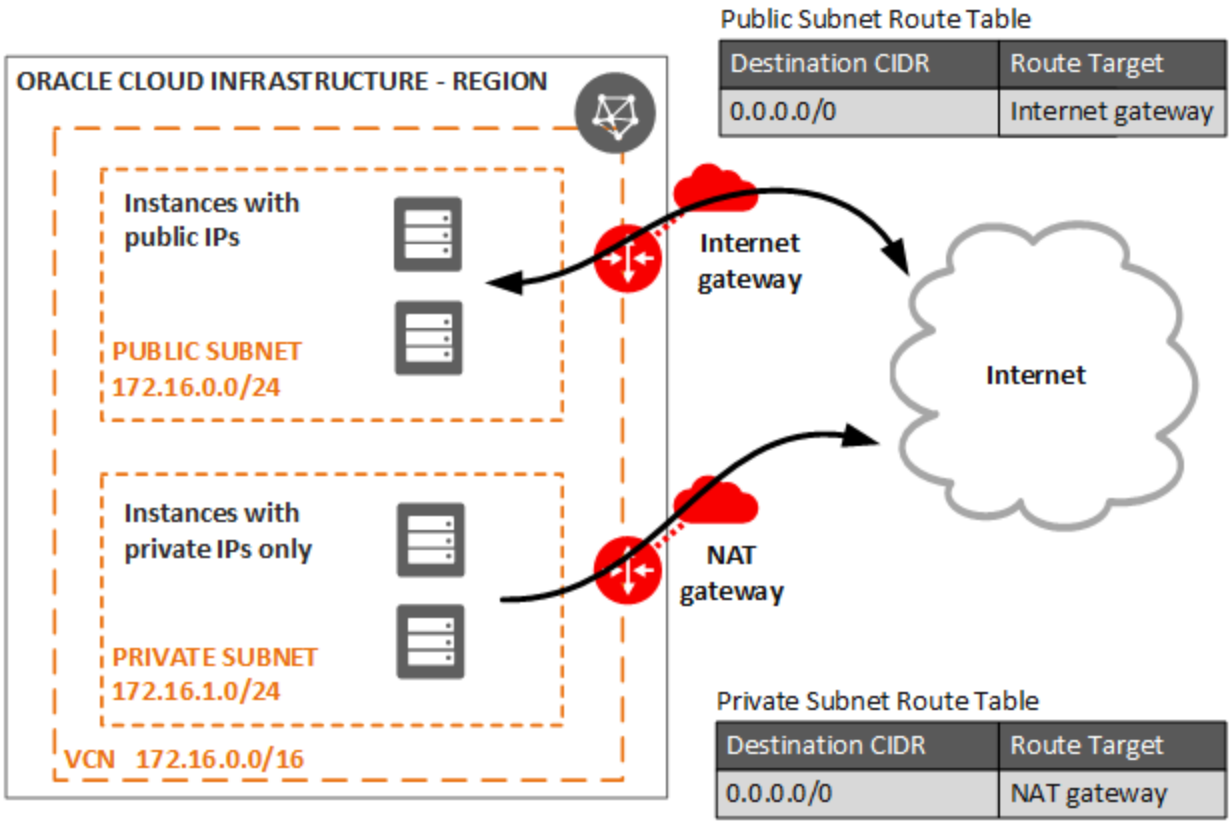
Example scenario: Imagine you have resources that need to receive inbound traffic from the internet (for example, web servers). You also have private resources that need to be protected from inbound traffic from the internet. All of these resources need to initiate connections to the internet to request software updates from sites on the internet.

You set up a VCN and add a public subnet to hold the web servers. When launching the instances, you assign public IP addresses to them so they can receive inbound internet traffic. You also add a private subnet to hold the private instances. They cannot have public IP addresses because they are in a private subnet.

You add an internet gateway to the VCN. You also add a route rule in the public subnet's route table that directs internet-bound traffic to the internet gateway. The public subnet's instances can now initiate connections to the internet and also receive inbound connections initiated from the internet. Remember that you can use a subnet's security lists to control the types of traffic that are allowed in and out of the instances at the packet level.

You add a NAT gateway to the VCN. You also add a route rule in the private subnet's route table that directs internet-bound traffic to the NAT gateway. The private subnet's instances can now initiate connections to the internet. The NAT gateway allows responses, but it does not allow connections that are *initiated from the internet*. Without that NAT gateway, the private instances would instead need to be in the public subnet and have public IP addresses to get their software updates.

The following diagram illustrates the basic network layout for the example. The arrows indicate whether connections can be initiated in only one direction or both.





Note

A NAT gateway can be used only by resources in the gateway's own VCN. If the VCN is peered with another, resources in the other VCN cannot access the NAT gateway.

Also, resources in an on-premises network connected to the NAT gateway's VCN with FastConnect or an IPSec VPN cannot use the NAT gateway.

Here are a few basics about NAT gateways:

- The NAT gateway supports TCP, UDP, and ICMP ping traffic.
- The gateway supports a maximum of approximately 20,000 concurrent connections to a single destination address and port.
- The Networking service automatically assigns a public IP address to the NAT gateway. You can't choose the public IP address or use one of your reserved public IP addresses.
- There's a limit on the number of NAT gateways per VCN. See Service Limits.

Routing for a NAT Gateway

You control routing in your VCN at the subnet level, so you can specify which subnets in your VCN use a NAT gateway. You can have more than one NAT gateway on a VCN (although you must request an increase in your limits). For example, if you want an external application to distinguish traffic from the VCN's different subnets, you could set up a different NAT gateway (and thus a different public IP address) for each subnet. A given subnet can route traffic to only a single NAT gateway.

Blocking Traffic Through a NAT Gateway

You create a NAT gateway in the context of a specific VCN. In other words, the NAT gateway is automatically always attached to only one VCN of your choice. However, you can block or allow traffic through the NAT gateway at any time. By default, the gateway allows traffic upon

creation. Blocking the NAT gateway prevents all traffic from flowing, regardless of any existing route rules or security lists in your VCN. For instructions on how to block traffic, see To block/allow traffic for a NAT gateway.

Transitioning to a NAT Gateway

If you're switching from using a NAT instance in your VCN to a NAT gateway, consider that the public IP address for your NAT device will change.

If you're switching from using an internet gateway to a NAT gateway, the instances with access to the NAT gateway no longer need public IP addresses to reach the internet. Also, the instances no longer need to be in a public subnet. You can't switch a subnet from public to private. However, you can delete the ephemeral public IPs from your instances if you like.

Deleting a NAT Gateway

To delete a NAT gateway, its traffic does not have to be blocked, but there must not be a route table that lists it as a target. For instructions, see To delete a NAT gateway.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: see IAM Policies for Networking.

Setting Up a NAT Gateway

Task 1: Create the NAT gateway

1. In the Console, confirm you're viewing the compartment that contains the VCN that you

want to add the NAT gateway to. For information about compartments and access control, see Access Control.

2. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
3. Click the VCN you're interested in.
4. Under **Resources**, click **NAT Gateways**.
5. Click **Create NAT Gateway**.
6. Enter the following values:
   - **Name:** A friendly name for the NAT gateway. It doesn't have to be unique. Avoid entering confidential information.
   - **Create in compartment**: The compartment where you want to create the NAT gateway, if different from the compartment you're currently working in.
   - **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
7. Click **Create NAT Gateway**.
   The NAT gateway is then created and displayed on the **NAT Gateways** page in the compartment you chose. The gateway allows traffic by default. At any time, you can block or allow traffic through it.

Task 2: Update routing for the subnet

When you create a NAT gateway, you must also create a route rule that directs the desired traffic from the subnet to the NAT gateway. You do this for each subnet that needs to access the gateway.

1. Determine which subnets in your VCN need access to the NAT gateway.
2. For each of those subnets, update the subnet's route table to include a new rule:
    a. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
    b. Click the VCN you're interested in.
    c. Under **Resources**, click **Route Tables**.
    d. Click the route table you're interested in.
    e. Click **Add Route Rule** and enter the following values:
        - **Target Type:** NAT Gateway.
        - **Destination CIDR Block:** 0.0.0.0/0
        - **Compartment:** The compartment where the NAT gateway is located.
        - **Target NAT Gateway:** The NAT gateway.
    f. Click **Add Route Rule**.

Any subnet traffic with a destination that matches the rule is routed to the NAT gateway. For more information about setting up route rules, see Route Tables.

Later, if you no longer need the NAT gateway and want to delete it, you must first delete all the route rules in your VCN that specify the NAT gateway as the target.



Tip

Without the required routing, traffic doesn't flow over the NAT gateway. If a situation occurs where you need to temporarily stop the traffic flow over the gateway, you can simply remove the route rule that enables traffic. Or you can block traffic through the gateway entirely. You do not need to delete it.

Using the Console

To create a NAT gateway

See the instructions in Task 1: Create the NAT gateway.

To block/allow traffic for a NAT gateway

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Under **Resources**, click **NAT Gateways**.
4. For the NAT gateway you're interested in, click the Actions icon (three dots) and then click **Block Traffic** (or **Allow Traffic** if you're enabling traffic for the NAT gateway).
5. Confirm when prompted.
   When the traffic is blocked, the NAT gateway's icon turns gray, and the label changes to BLOCKED. When the traffic is allowed, the NAT gateway's icon turns green, and the label changes to AVAILABLE.

To update a NAT gateway

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Under **Resources**, click **NAT Gateways**.
4. For the NAT gateway you're interested in, click the Actions icon (three dots), and then click **Edit**.
5. Make your changes and click **Save Changes**.

To delete a NAT gateway

Prerequisite: There must not be a route table that lists the NAT gateway as a target.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Under **Resources**, click **NAT Gateways**.
4. For the NAT gateway you want to delete, click the Actions icon (three dots), and then click **Terminate**.
5. Confirm when prompted.

To manage tags for a NAT gateway

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Under **Resources**, click **NAT Gateways**.
4. Click the Actions icon (three dots) for the NAT gateway, and then click **View Tags**. From there you can view the existing tags, edit them, and apply new ones.

For more information, see Resource Tags.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

To manage your NAT gateways, use these operations:

- ListNatGateways
- CreateNatGateway

- GetNatGateway
- UpdateNatGateway
- DeleteNatGateway

To manage route tables, see Route Tables.

Access to Oracle Services: Service Gateway

This topic describes how to set up and manage a service gateway. A service gateway enables cloud resources without public IP addresses to privately access Oracle services.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Access to Oracle Services

The *Oracle Services Network* is a conceptual network in Oracle Cloud Infrastructure that is reserved for Oracle services. These services have public IP addresses that can be reached over the internet. However, you can access the Oracle Services Network public IP addresses without the traffic going over the internet. There are different ways, depending on which of your hosts need the access:

- **Hosts in your on-premises network:** Use FastConnect public peering.
- **Resources in your VCN:** Use a service gateway.

Highlights

- A service gateway lets your virtual cloud network (VCN) privately access specific Oracle services without exposing the data to the public internet. No internet gateway or NAT is required to reach those specific services. The resources in the VCN can be in a private subnet and use only private IP addresses. The traffic from the VCN to the Oracle service travels over the Oracle network fabric and never traverses the internet.
- The service gateway is regional and enables access only to supported Oracle services *in the same region* as the VCN.
- The service gateway allows access to supported Oracle services within the region to protect your data from the internet. Your workloads may require access to public endpoints or services not supported by the service gateway (for example, to download updates or patches). Ensure you have a NAT gateway or other access to the internet if necessary.
- The supported Oracle services are Oracle Cloud Infrastructure Object Storage and others in the Oracle Services Network. For a list, see Service Gateway: Supported Cloud Services in Oracle Services Network.
- The service gateway uses the concept of a *service CIDR label*, which is a string that represents all the regional public IP address ranges for the service or group of services of interest (for example, *OCI PHX Object Storage* is the string for Object Storage in us-phoenix-1). You use that service CIDR label when you configure the service gateway and related route rules to control traffic to the service. You can optionally use it when configuring security list rules. If the service's public IP addresses change in the future, you don't have to adjust those rules.

Overview of Service Gateways

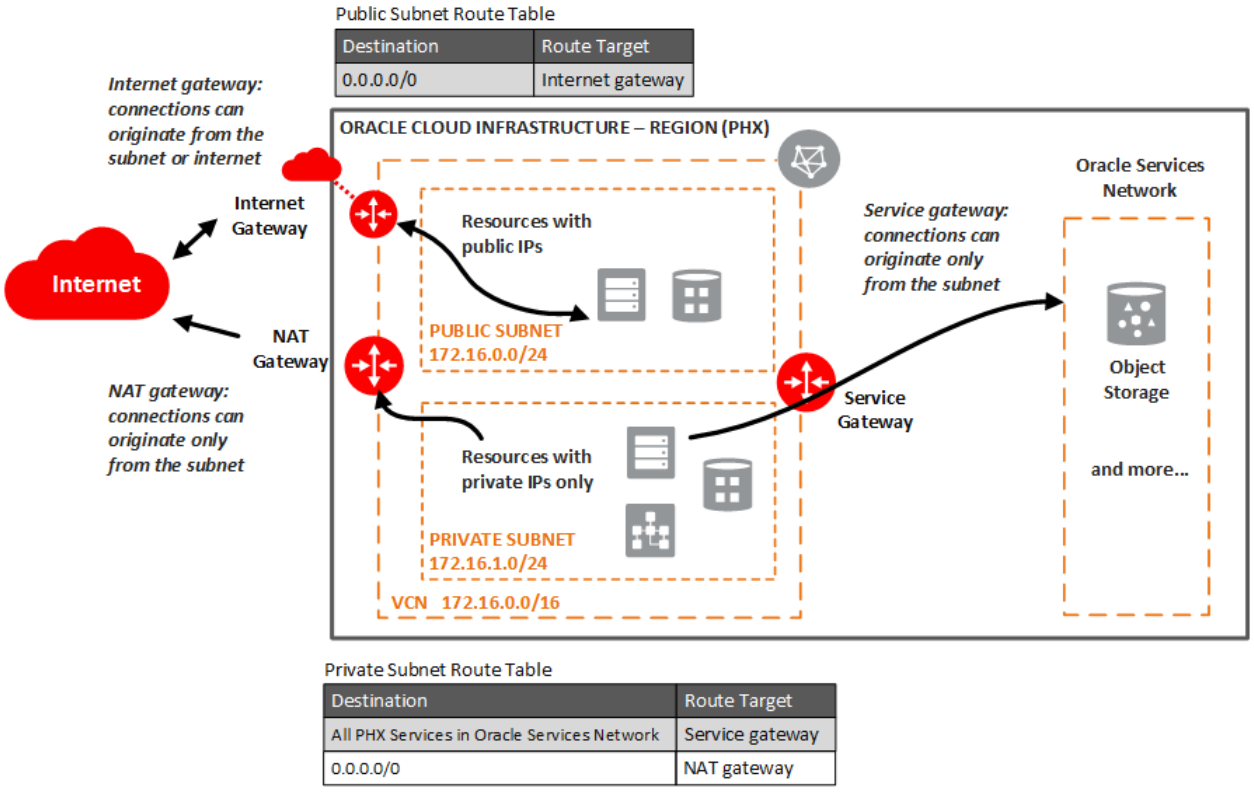
A service gateway lets resources in your VCN privately access specific Oracle services, without exposing the data to an internet gateway or NAT. The resources in the VCN can be in a private subnet and use only private IP addresses. The traffic from the VCN to the service of interest travels over the Oracle network fabric and never traverses the internet.

The following simple diagram illustrates a VCN that has both a public subnet and a private subnet. Resources in the private subnet have only private IP addresses.

The VCN has three gateways:

- **Internet gateway:** To provide the public subnet direct access to public endpoints on the internet. Connections can be initiated from the subnet or from the internet. The resources in the public subnet must have public IP addresses. For more information, see Internet Gateway.
- **Service gateway:** To provide the private subnet with private access to supported Oracle services within the region. Connections can be initiated only from the subnet.
- **NAT gateway:** To provide the private subnet with private access to public endpoints on the internet. Connections can be initiated only from the subnet. For more information, see NAT Gateway.

You control routing in your VCN at the subnet level, so you can specify which subnets in your VCN use each gateway. In the diagram, the route table for the public subnet sends non-local traffic through the internet gateway. The route table for the private subnet sends traffic destined for the Oracle services through the service gateway. It sends all remaining traffic to the NAT gateway.



Important

See this known issue for information about configuring route rules with *service gateway* as the target on route tables associated with public subnets.

A service gateway can be used only by resources in the gateway's own VCN. If the VCN is peered with another, resources in the other VCN cannot access the service gateway. Also, resources in an on-premises network connected to the service gateway's VCN with FastConnect or an IPSec VPN cannot use the service gateway. However, the on-premises network can use FastConnect public peering for private access to public Oracle Cloud

Infrastructure services. For a list of services available with FastConnect public peering, see FastConnect Supported Cloud Services.

A VCN can have only one service gateway. For more information about limits, see Service Limits.

For instructions on setting up a service gateway, see Setting Up a Service Gateway in the Console.

About Service CIDR Labels

Each Oracle service has a regional public endpoint that uses public IP addresses for access. When you set up a service gateway with access to an Oracle service, you also set up Networking service route rules and optionally security list rules that control traffic with the service. That means you need to know the service's public IP addresses to set up those rules. To make it easier for you, the Networking service uses *service CIDR labels* to represent all the public CIDRs for a given Oracle service or a group of Oracle services. If a service's CIDRs change in the future, you don't have to adjust your route rules or security list rules.

Examples:

- **OCI PHX Object Storage** is a service CIDR label that represents all the Object Storage CIDRs in the us-phoenix-1 region.
- **All PHX Services in Oracle Services Network** is a service CIDR label that represents all the CIDRs for the supported services in the Oracle Services Network in the us-phoenix-1 region. For a list of the services, see Service Gateway: Supported Cloud Services in Oracle Services Network.

As you can see, a service CIDR label can be associated with either a single Oracle service (example: Object Storage), or multiple Oracle services.

The term *service* is often used in this topic in place of the more accurate term *service CIDR label*. The important thing to remember is that when you set up a service gateway (and related route rules), you specify the *service CIDR label* you're interested in. In the Console, you're presented with the available service CIDR labels. If you use the REST API, the ListServices operation returns the available `Service` objects. The `Service` object's

`cidrBlock` attribute contains the service CIDR label (example: `all-phx-services-in-oracle-services-network`).

Available Service CIDR Labels

Here are the available service CIDR labels:

- **OCI *<region>* Object Storage:** For information about the service, see Overview of Object Storage
- **All *<region>* Services in Oracle Services Network:** For a list of supported services, see Service Gateway: Supported Cloud Services in Oracle Services Network.

> **Important**
>
> See this known issue for information about accessing Oracle YUM services through the service gateway.

Enabling a Service CIDR Label for a Service Gateway

To give your VCN access to a given service CIDR label, you must *enable* that service CIDR label for the VCN's service gateway. You can do that when you create the service gateway, or later after it's created. You can also disable a service CIDR label for the service gateway at any time.



Important

Because Object Storage is covered by both **OCI *<region>* Object Storage** and **All *<region>* Services in Oracle Services Network**, a service gateway can use **only one of those service CIDR labels**. Likewise, a route table can have a single rule for one of the service CIDR labels. It cannot have two separate rules, one for each label.

If the service gateway is configured to use **All *<region>* Services in Oracle Services Network**, the route rule can use either CIDR label. However, if the service gateway is configured to use **OCI *<region>* Object Storage** and the route rule uses **All *<region>* Services in Oracle Services Network**, traffic to services in the Oracle Services Network except Object Storage will be blackholed. The Console prohibits you from configuring the service gateway and corresponding route table in that manner.

If you want to switch the service gateway to use a different service CIDR label, see To switch to a different service CIDR label.

Blocking Traffic Through a Service Gateway

You create a service gateway in the context of a specific VCN. In other words, the service gateway is always attached to that one VCN. However, you can block or allow traffic through the service gateway at any time. By default, the gateway allows traffic flow upon creation. Blocking the service gateway traffic prevents all traffic from flowing, regardless of what service CIDR labels are enabled, or any existing route rules or security lists in your VCN. For instructions on how to block traffic, see To block or allow traffic for a service gateway.

Route Rules and Security List Rules for a Service Gateway

For traffic to be routed from a subnet in your VCN to a service gateway, you must add a rule accordingly to the subnet's route table. The rule must use the service gateway as the target. For the destination, you must use the service CIDR label that is enabled for the service gateway. This means that you don't have to know the specific public CIDRs, which could change over time.

Any traffic leaving the subnet and destined for the service's public CIDRs is then routed to the service gateway. If the service gateway traffic is blocked, no traffic flows through it even if there's a route rule that matches the traffic. For instructions on setting up route rules for a service gateway, see Task 2: Update routing for the subnet.

The subnet's security lists must also allow the desired traffic. If you like, you can use a service CIDR label instead of a CIDR for the source or destination of the desired traffic. Again, this means that you don't have to know the specific public CIDRs for the service. For convenience, you can use a service CIDR label in security list rules even if your VCN doesn't have a service gateway, and the traffic to the services uses an internet gateway.

You can use stateful or stateless security list rules that use a service CIDR label:

- **For stateful rules:** Create an egress rule with the destination service = the service CIDR label of interest. As with any security list rule, you can specify other items such as the IP protocol and source and destination ports.
- **For stateless rules:** You must have both egress and ingress rules. Create an egress rule with the destination service = the service CIDR label of interest. Also create an ingress rule with the source service = the service CIDR label of interest. As with any security list rule, you can specify other items such as the IP protocol and source and destination ports.

For instructions on setting up security list rules that use a service CIDR label, see Task 3: (Optional) Update the security lists for the subnet.

Object Storage: Allowing Bucket Access from Only a Particular VCN or CIDR Range

If you use a service gateway to access Object Storage, you can write an IAM policy that allows access to a particular Object Storage bucket only if these requirements are met:

- The request goes through a service gateway.
- The request originates from the particular VCN or CIDR (for example, a subnet within a VCN) that is specified in the policy.

For examples of this particular type of IAM policy, and important caveats about its use, see Task 4: (Optional) Update IAM Policies to Restrict Object Storage Bucket Access.

Deleting a Service Gateway

To delete a service gateway, its traffic does not have to be blocked, but there must not be a route table that lists it as a target. See To delete a service gateway.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: see IAM Policies for Networking.

Setting Up a Service Gateway in the Console

Following is the process for setting up a service gateway. It assumes that you already have a VCN with a subnet (either private or public).



Important

The service gateway allows access to supported Oracle services within the region to protect your data from the internet. Your applications may require access to public endpoints or services not supported by the service gateway (for example, to download updates or patches). Ensure you have a NAT gateway or other access to the internet if necessary.

Task 1: Create the service gateway

1. In the Console, confirm you're viewing the compartment that contains the VCN that you want to add the service gateway to. For information about compartments and access control, see Access Control.
2. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
3. Click the VCN you're interested in.
4. On the left side of the page, click **Service Gateways**.
5. Click **Create Service Gateway**.
6. Enter the following values:
   - **Name:** A descriptive name for the service gateway. It doesn't have to be unique. Avoid entering confidential information.
   - **Create in compartment**: The compartment where you want to create the service gateway, if different from the compartment you're currently working in.
   - **Services:** Optionally select the service CIDR label you're interested in. If you don't select one now, you can later update the service gateway and add a service CIDR label then. Without at least one service CIDR label enabled for the gateway, no traffic flows through it.

- **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

7. Click **Create Service Gateway**.
   The service gateway is then created and displayed on the **Service Gateways** page in the compartment you chose. The gateway allows traffic through it by default. At any time, you can block or allow the traffic through it.

Task 2: Update routing for the subnet

When you configure a service gateway for a particular service CIDR label, you must also create a route rule that specifies that label as the destination and the target as the service gateway. You do this for each subnet that needs to access the gateway.

1. Determine which subnets in your VCN need access to the service gateway.
2. For each of those subnets, update the subnet's route table to include a new rule:
   a. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
   b. Click the VCN you're interested in.
   c. Under **Resources**, click **Route Tables**.
   d. Click the route table you're interested in.
   e. Click **Edit Route Rules.**
   f. Click **Add Route Rule** and enter the following values:
      - **Target Type:** Service Gateway.
      - **Destination Service:** The service CIDR label that is enabled for the gateway.

- **Compartment:** The compartment where the service gateway is located.
- **Target:** The service gateway.

g. Click **Save**.

Any subnet traffic with a destination that matches the rule is routed to the service gateway. For more information about setting up route rules, see Route Tables.

Later, if you no longer need the service gateway and want to delete it, you must first delete all the route rules in your VCN that specify the service gateway as the target.



Tip

Without the required routing, traffic doesn't flow over the service gateway. If a situation occurs where you want to temporarily stop the traffic flow over the gateway to a particular service, you can simply remove the route rule that enables traffic. You can also disable that particular service CIDR label for the gateway. Or you can block all traffic through the service gateway entirely. You do not have to delete the gateway.

Task 3: (Optional) Update the security lists for the subnet

When you configure a service gateway to access a service CIDR label, you must also ensure that the security lists are configured to allow the desired traffic. Your security lists might already allow this traffic, which is why this task is optional. The following procedure describes how to set up a rule that uses the service CIDR label. You do this for each subnet that needs to access the gateway.

1. Determine which subnets in your VCN need to connect to the services you're interested in.
2. Update the security list for each of those subnets to include rules to allow the desired

egress or ingress traffic with the particular service:

a. In the Console, while viewing the VCN you're interested in, click **Security Lists**.
b. Click the security list you're interested in.
c. Click **Edit All Rules** and create one or more rules, each for the specific type of traffic you want to allow. See the following example for more details.

Example

Let's say you want to add a stateful rule that enables egress HTTPS (TCP port 443) traffic from the subnet to both Object Storage and Oracle YUM repos. Here are the basic steps you take when adding a rule:

i. In the **Allow Rules for Egress** section, click **+Add Rule**.
ii. Leave the **Stateless** check box unselected.
iii. **Destination Type:** Service.
iv. **Destination Service:** The service CIDR label that you're interested in. To access both Object Storage and Oracle YUM repos, it's **All *<region>* Services in Oracle Services Network**.
v. **IP Protocol:** Leave as TCP.
vi. **Source Port Range:** Leave as All.
vii. **Destination Port Range**: Enter 443.

d. Click **Save Security List Rules** at the bottom of the dialog box.

For more information about setting up security list rules, see Security Lists.

Task 4: (Optional) Update IAM Policies to Restrict Object Storage Bucket Access

This task is applicable only if you're using a service gateway to access Object Storage. You can optionally write an IAM policy to allow only the resources in a specific VCN to write

objects to a particular bucket.



Important

If you use one of the following IAM policies to restrict access to a bucket, the bucket is *not accessible from the Console*. It's accessible only from within the specific VCN or CIDR block.

Also, the IAM policies allow requests to Object Storage only if they go from the specified VCN or CIDR block *through the service gateway*. If they go through the internet gateway, the requests are denied.

The following example lets resources in the example ObjectBackup group write objects to an existing bucket called db-backup that resides in a compartment called ABC.

```
Allow group ObjectBackup to read buckets in compartment ABC

Allow group ObjectBackup to manage objects in compartment ABC where
   all {target.bucket.name='db-backup',
        request.vcn.id='<VCN_OCID>',
        any {request.permission='OBJECT_CREATE', request.permission='OBJECT_INSPECT'}}
```

Instead of specifying a VCN's OCID, you could specify a CIDR block. For example, the next version of the policy includes a `request.ipv4.ipaddress` variable with a value of 10.0.1.0/24. If the resource making the request has an IP address that is not in this CIDR block, the request is denied.

```
Allow group ObjectBackup to read buckets in compartment ABC

Allow group ObjectBackup to manage objects in compartment ABC where
   all {target.bucket.name='db-backup',
        request.vcn.id='<VCN_OCID>',
        request.ipv4.ipaddress in ['10.0.1.0/24'],
        any {request.permission='OBJECT_CREATE', request.permission='OBJECT_INSPECT'}}
```

Managing a Service Gateway in the Console

To create a service gateway

See the instructions in Task 1: Create the service gateway.

To switch to a different service CIDR label

To avoid disrupting your Object Storage connections while switching between the **OCI *<region>* Object Storage** service CIDR label and **All *<region>* Services in Oracle Services Network**, use the following process:

1. Update the service gateway: Remove the existing service CIDR label, and then add the one you want to switch to. You can't enable both labels for the service gateway.
2. Update relevant route rules: For each rule that uses the service gateway as the target, switch the rule's destination service from the existing service CIDR label to the one you want to switch to.
3. Update relevant security list rules: Change any security list rules that specify the existing service CIDR label to instead use the one you want to switch to.

If you instead delete your existing service gateway and create a new one, your Object Storage connections will be disrupted. Remember that before you can delete a service gateway, you must delete any route rules that specify that gateway as a target.

To update a service gateway

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Click **Service Gateways**.
4. For the service gateway you're interested in, click the Actions icon (three dots), and then click **Edit**.

5. Make your changes and click **Save**.

To block or allow traffic for a service gateway

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Click **Service Gateways**.
4. For the service gateway you're interested in, click the Actions icon (three dots), and then click **Block Traffic** (or **Allow Traffic** if you're enabling traffic for the service gateway).
   When the traffic is blocked, the service gateway's icon turns gray, and the label changes to BLOCKED. When the traffic is allowed, the service gateway's icon turns green, and the label changes to AVAILABLE.

To delete a service gateway

Prerequisite: The service gateway does not have to block traffic, but there must not be a route table that lists it as a target.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Click **Service Gateways**.
4. For the service gateway you're interested in, click the Actions icon (three dots), and then click **Delete**.
5. Confirm when prompted.

To manage tags for a service gateway

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. For the service gateway you're interested in, click the Actions icon (three dots), and then click **View Tags**. From there you can view the existing tags, edit them, and apply new ones.

For more information, see Resource Tags.

Managing a Service Gateway with the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.



Warning

If anyone in your organization implements a service gateway, be aware that you **may need to update any client code that works with Networking service route rules and security lists**. There are possible breaking API changes. For more information, see the service gateway release notes.

To manage your service gateways, use these operations:

- ListServiceGateways
- CreateServiceGateway
- GetServiceGateway
- UpdateServiceGateway

- DeleteServiceGateway
- ListServices: Use this to determine the available service CIDR labels.
- GetService: Gets the details for a particular service CIDR label.
- AttachServiceId: Enables a service CIDR label for a service gateway.
- DetachServiceId: Disables a service CIDR label for a service gateway.

To manage route tables, see Route Tables. To manage security lists, see Security Lists. To manage IAM policies, see Managing Policies.

Access to Other VCNs: Peering

VCN peering is the process of connecting multiple virtual cloud networks (VCNs). There are two types of VCN peering:

- Local VCN peering (within region)
- Remote VCN peering (across regions)

You can use VCN peering to divide your network into multiple VCNs (for example, based on departments or lines of business), with each VCN having direct, private access to the others. There's no need for traffic to flow over the internet or through your on-premises network by way of an IPSec VPN or FastConnect. You can also place shared resources into a single VCN that all the other VCNs can access privately.

Because remote VCN peering crosses regions, you can use it (for example) to mirror or back up your databases in one region to another.

Important Implications of Peering

This section summarizes some access control, security, and performance implications for peered VCNs. In general, you can control access and traffic between two peered VCNs by using IAM policies, route tables in each VCN, and security lists in each VCN.

Controlling the Establishment of Peerings

With IAM policies, you can control:

- Who can subscribe your tenancy to another region (required for remote VCN peering).
- Who in your organization has the authority to establish VCN peerings (for example, see the IAM policies in Setting Up a Local Peering and Setting Up a Remote Peering). Be aware that deletion of these IAM policies does not affect any existing peerings, only the ability for future peerings to be created.
- Who can manage route tables and security lists.

Controlling Traffic Flow Over the Connection

Even if a peering connection has been established between your VCN and another, you can control the packet flow over the connection with route tables in your VCN. For example, you can restrict traffic to only specific subnets in the other VCN.

Without terminating the peering, you can stop traffic flow to the other VCN by simply removing route rules that direct traffic from your VCN to the other VCN. You can also effectively stop the traffic by removing any security list rules that enable ingress or egress traffic with the other VCN. This doesn't stop traffic flowing over the peering connection, but stops it at the VNIC level.

For more information about the routing and security lists, see the discussions in these sections:

Local VCN peering:

- Important Local Peering Concepts
- Task E: Configure the route tables
- Task F: Configure the security lists

Remote VCN peering:

- Important Remote Peering Concepts
- Task E: Configure the route tables

- Task F: Configure the security lists

Controlling the Specific Types of Traffic Allowed

It's important that each VCN administrator ensure that all outbound and inbound traffic with the other VCN is intended/expected and well defined. In practice, this means implementing security list rules that explicitly state the types of traffic your VCN can send to the other and accept from the other.



Important

Your instances running Oracle-provided Linux images or Windows images also have firewall rules that control access to the instance. When troubleshooting access to an instance, make sure both the security lists associated with the instance's subnet and the instance's firewall rules are set correctly. For more information, see Oracle-Provided Images.

If your instance is running Oracle Linux 7, you need to use firewalld to interact with the iptables rules. For your reference, here are commands for opening a port (1521 in this example):

```
sudo firewall-cmd --zone=public --permanent --add-
port=1521/tcp

sudo firewall-cmd --reload
```

In addition to security lists and firewalls, you should evaluate other OS-based configuration on the instances in your VCN. There could be default configurations that don't apply to your own VCN's CIDR, but inadvertently apply to the other VCN's CIDR.

Using Default Security List Rules

If your VCN's subnets use the default security list with the default rules it comes with, be aware that there are two rules that allow ingress traffic from anywhere (that is, 0.0.0.0/0, and thus the other VCN):

- Stateful ingress rule that allows TCP port 22 (SSH) traffic from 0.0.0.0/0 and any source port
- Stateful ingress rule that allows ICMP type 3, code 4 traffic from 0.0.0.0/0 and any source port

Make sure to evaluate these rules and whether you want to keep or update them. As stated earlier, you should ensure that all inbound or outbound traffic that you permit is intended/expected and well defined.

Preparing for Performance Impact and Security Risks

In general, you should prepare your VCN for the ways it could be affected by the other VCN. For example, the load on your VCN or its instances could increase. Or your VCN could experience a malicious attack directly from or by way of the other VCN.

Regarding performance: If your VCN is providing a service to another, be prepared to scale up your service to accommodate the demands of the other VCN. This might mean being prepared to launch additional instances as necessary. Or if you're concerned about high levels of network traffic coming to your VCN, consider using stateless security list rules to limit the level of connection tracking your VCN must perform. Stateless security list rules can also help slow the impact of a denial-of-service (DoS) attack.

Regarding security risks: You can't necessarily control whether the other VCN is connected to the internet. If it is, be aware that your VCN can be exposed to bounce attacks in which a malicious host on the internet can send traffic to your VCN but make it look like it's coming from the VCN you're peered with. To guard against this, as mentioned earlier, use your security lists to carefully limit the inbound traffic from the other VCN to expected and well-defined traffic.

Local VCN Peering (Within Region)

This topic is about *local VCN peering.* In this case, *local* means that the VCNs reside *in the same region.* If the VCNs are in different regions, see Remote VCN Peering (Across Regions).



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Overview of Local VCN Peering

Local VCN peering is the process of connecting two VCNs in the same region so that their resources can communicate using private IP addresses without routing the traffic over the internet or through your on-premises network. The VCNs can be in the same Oracle Cloud Infrastructure tenancy or different ones. Without peering, a given VCN would need an internet gateway and public IP addresses for the instances that need to communicate with another VCN.

For more information, see Access to Other VCNs: Peering.

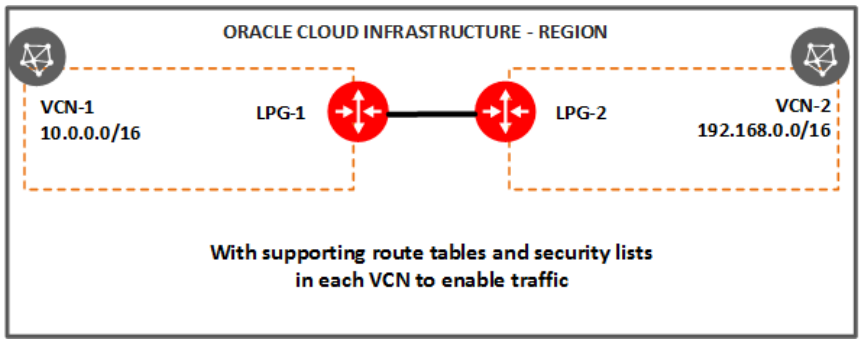
Summary of Networking Components for Peering

At a high level, the Networking service components required for a local peering include:

- Two VCNs with non-overlapping CIDRs, in the same region
- A *local peering gateway (LPG)* on each VCN in the peering relationship.
- A *connection* between those two LPGs.
- Supporting route rules to enable traffic to flow over the connection, and only to/from select subnets in the respective VCNs (if desired).

- Supporting security list rules to control the types of traffic allowed to/from the instances in the subnets that need to communicate with the other VCN.

The following diagram illustrates the components.





Note

A given VCN can use the peered LPGs to reach these resources:

- VNICs in the other VCN
- An on-premises network attached to the other VCN, if an advanced routing scenario called transit routing has been set up for the VCNs

A VCN can't use its peered VCN to reach other destinations outside of the VCNs (such as the internet). For example, if VCN-1 in the preceding diagram were to have an internet gateway, the instances in VCN-2 could not use it to send traffic to endpoints on the internet. However, be aware that VCN-2 could *receive* traffic from the internet by way of VCN-1. For more information, see Important Implications of VCN Peering.

Explicit Agreement Required from Both Sides

Peering involves two VCNs that might be owned by the same party or two different ones. The two parties might both be in your company but in different departments. Or the two parties might be in entirely different companies (for example, in a service-provider model).

Peering between two VCNs requires explicit agreement from both parties in the form of Oracle Cloud Infrastructure Identity and Access Management policies that each party implements for their own VCN's compartment or tenancy. If the VCNs are in different tenancies, each administrator must provide their tenancy OCID and put in place special policy statements to enable the peering.

Advanced Scenario: Transit Routing

There's an advanced routing scenario called transit routing that enables communication between an on-premises network and multiple VCNs over a *single* Oracle Cloud Infrastructure FastConnect or IPSec VPN. The VCNs must be in the same region and locally peered in a hub-and-spoke layout. As part of the scenario, the VCN that is acting as the hub has a route table *associated with each LPG* (typically route tables are associated with a VCN's subnets).

When you create an LPG, you can optionally associate a route table with it. Or if you already have an existing LPG without a route table, you can associate a route table with it. The route table must belong to the LPG's VCN. A route table associated with an LPG can contain only rules that use the VCN's attached DRG as the target.

An LPG can exist without a route table associated with it. However, after you associate a route table with an LPG, there must always be a route table associated with it. But, you can associate a *different* route table. You can also edit the table's rules, or delete some or all of the rules.

Important Local Peering Concepts

The following concepts help you understand the basics of VCN peering and how to establish a local peering.

PEERING

A *peering* is a single peering relationship between two VCNs. Example: If VCN-1 peers with three other VCNs, then there are three peerings. The *local* part of *local peering* indicates that the VCNs are in the same region. A given VCN can have a maximum of ten local peerings at a time.



Warning

The two VCNs in the peering relationship must not have overlapping CIDRs. However, if VCN-1 is peered with three other VCNs, those three VCNs can have overlapping CIDRs with each other. You would set up the subnets in VCN-1 to have route rules that direct traffic to the targeted peered VCN.

VCN ADMINISTRATORS

In general, VCN peering can occur only if both of the VCN administrators agree to it. In practice, this means that the two administrators must:

- Share some basic information with each other.
- Coordinate to set up the required Oracle Cloud Infrastructure Identity and Access Management policies to enable the peering.
- Configure their VCNs for the peering.

Depending on the situation, a single administrator might be responsible for both VCNs and the related policies.

For more information about the required policies and VCN configuration, see Setting Up a Local Peering.

ACCEPTOR AND REQUESTOR

To implement the IAM policies required for peering, the two VCN administrators must designate one administrator as the *requestor* and the other as the *acceptor*. The requestor must be the one to initiate the request to connect the two LPGs. In turn, the acceptor must create a particular IAM policy that gives the requestor permission to connect to LPGs in the acceptor's compartment. Without that policy, the requestor's request to connect fails.

LOCAL PEERING GATEWAY (LPG)

A *local peering gateway (LPG)* is a component on a VCN for routing traffic to a locally peered VCN. As part of configuring the VCNs, each administrator must create an LPG for their VCN. A given VCN must have a separate LPG for each local peering it establishes (maximum ten LPGs per VCN). To continue with the previous example: VCN-1 would have three LPGs to peer with three other VCNs. In the API, a LocalPeeringGateway is an object that contains information about the peering. You can't reuse an LPG to later establish another peering.

PEERING CONNECTION

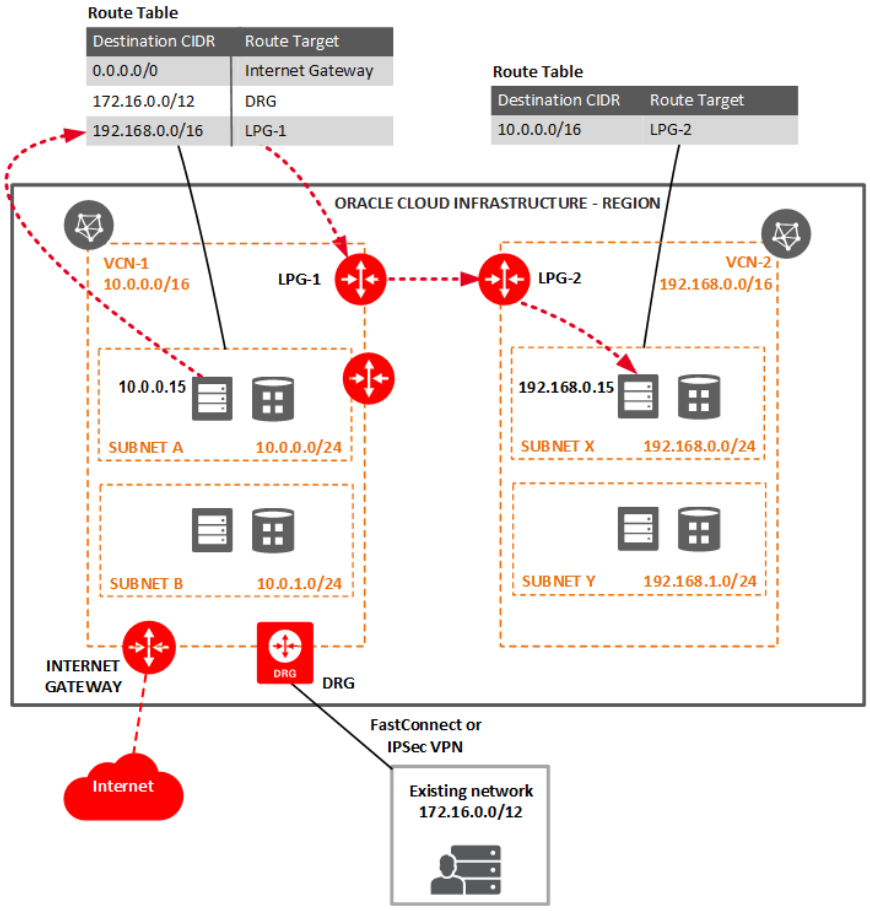
When the requestor initiates the request to peer (in the Console or API), they're effectively asking to *connect the two LPGs*. The requestor must have information to identify each LPG (such as the LPG's compartment and name, or the LPG's OCID). Each administrator must put the required IAM policies in place for their compartment or tenancy.

Either VCN administrator can terminate a peering by deleting their LPG. In that case, the other LPG's status switches to REVOKED. The administrator could instead render the connection non-functional by removing the route rules or security lists that enable traffic to flow across the connection (see the next sections).

ROUTING TO THE LPG

As part of configuring the VCNs, each administrator must update the VCN's routing to enable traffic to flow between the VCNs. In practice, this looks just like routing you set up for any gateway (such as an internet gateway or dynamic routing gateway). For each subnet that needs to communicate with the other VCN, you update the subnet's route table. The route rule specifies the destination traffic's CIDR and your LPG as the target. Your LPG routes traffic that matches that rule to the other LPG, which in turn routes the traffic to the next hop in the other VCN.

In the following diagram, VCN-1 and VCN-2 are peered. Traffic from an instance in Subnet A (10.0.0.15) that is destined for an instance in VCN-2 (192.168.0.15) is routed to LPG-1 based on the rule in Subnet A's route table. From there the traffic is routed to LPG-2, and then from there, on to its destination in Subnet X.



Note

As mentioned earlier, a given VCN can use the peered LPGs to reach VNICs in the other VCN, or the on-premises network if transit routing is set up for



the VCNs. But a VCN can't use the peered VCN to reach other destinations outside of the VCNs (such as the internet). For example, in the preceding diagram, VCN-2 cannot use the internet gateway attached to VCN-1.

SECURITY LIST RULES

Each subnet in a VCN has one or more security lists that control traffic in and out of the subnet's VNICs at the packet level. You can use security lists to control the type of traffic allowed with the other VCN. As part of configuring the VCNs, each administrator must determine which subnets in their own VCN need to communicate with VNICs in the other VCN and update their subnet's security lists accordingly.

Important Implications of VCN Peering

If you haven't yet, read Important Implications of Peering to understand important access control, security, and performance implications for peered VCNs.

Setting Up a Local Peering

Here's the general process for setting up a peering between two VCNs in the same region:

A. **Create the LPGs:** Each VCN administrator creates an LPG for their own VCN.

B. **Share information:** The administrators share the basic required information.

C. **Set up the required IAM policies for the connection:** The administrators set up IAM policies to enable the connection to be established.

D. **Establish the connection:** The requestor connects the two LPGs.

E. **Update route tables:** Each administrator updates their VCN's route tables to enable traffic between the peered VCNs as desired.

F. **Update security lists:** Each administrator updates their VCN's security lists to enable traffic between the peered VCNs as desired.

If desired, the administrators can perform tasks E and F *before* establishing the connection. In that case, each administrator must know the CIDR block or specific subnets from the other's VCN and share that in task B. After the connection is established, you can also get the CIDR block of the other VCN by viewing your own LPG's details in the Console. Look for **Peer Advertised CIDR**. Or if you're using the API, see the `peerAdvertisedCidr` parameter.

Task A: Create the LPGs

Each administrator creates an LPG for their own VCN. "You" in the following procedure means an administrator (either the acceptor or requestor).



Note

Required IAM Policy to Create LPGs

If the administrators already have broad network administrator permissions (see Let network admins manage a cloud network), then they have permission to create, update, and delete LPGs. Otherwise, here's an example policy giving the necessary permissions to a group called LPGAdmins. The second statement is required because creating an LPG affects the VCN it belongs to, so the administrator must have permission to manage VCNs.

```
Allow group LPGAdmins to manage local-peering-gateways in
tenancy
Allow group LPGAdmins to manage vcns in tenancy
```

1. In the Console, confirm you're viewing the compartment that contains the VCN that you want to add the LPG to. For information about compartments and access control, see Access Control.
2. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.

3. Click the VCN you're interested in.
4. Under **Resources**, click **Local Peering Gateways**.
5. Click **Create Local Peering Gateway**.
6. Enter the following:
   - **Name:** A friendly name for the LPG. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.
   - **Create in compartment**: The compartment where you want to create the LPG, if different from the compartment you're currently working in.
   - **Associate with Route Table (optional)**: Only if you're setting up the advanced routing scenario called transit routing. Select the compartment that contains the route table you want to associate with the LPG, and the route table itself. You can skip this part and associate the LPG with a route table later if you like.
   - **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.
7. Click **Create Local Peering Gateway**.
   The LPG is then created and displayed on the **Local Peering Gateways** page in the compartment you chose.

Task B: Share information

If you're the **requestor**, give this information to the acceptor (for example, by email or other out-of-band method):

- If the VCNs are in the **same tenancy:** Name of the IAM group that should be granted permission to create a connection in the acceptor's compartment. In the example in the

next task, the group is RequestorGrp.

- If the VCNs are in **different tenancies:** OCID for your tenancy, and OCID for the IAM group that should be granted permission to create a connection in the acceptor's compartment. In the example in the next task, it's the OCID for the RequestorGrp.
- Optional: Your VCN's CIDR, or specific subnets for peering with the other VCN.

If you're the **acceptor**, give this information to the requestor:

- If the VCNs are in the **same tenancy:** OCID for your LPG. Optionally, also the names of your VCN, LPG, and the compartment each is in.
- If the VCNs are in **different tenancies:** OCID for your LPG, and OCID for your tenancy.
- Optional: Your VCN's CIDR, or specific subnets for peering with the other VCN.

Task C: Set up the IAM policies (VCNs in same tenancy)

In this version of task C, both VCNs are in the same tenancy. If they're in different tenancies, instead see Task C: Set up the IAM policies (VCNs in different tenancies) .

Both the requestor and acceptor must ensure that the right policies are in place:

- **Policy R (implemented by the requestor):**

  ```
  Allow group RequestorGrp to manage local-peering-from in compartment RequestorComp
  ```

  The requestor is in an IAM group called RequestorGrp. This policy lets anyone in the group initiate a connection from any LPG in the requestor's compartment (RequestorComp). Policy R can be attached to either the tenancy (root compartment) or to RequestorComp. For information about why you would attach it to one versus the other, see Policy Attachment.

- **Policy A (implemented by the acceptor):**

  ```
  Allow group RequestorGrp to manage local-peering-to in compartment AcceptorComp

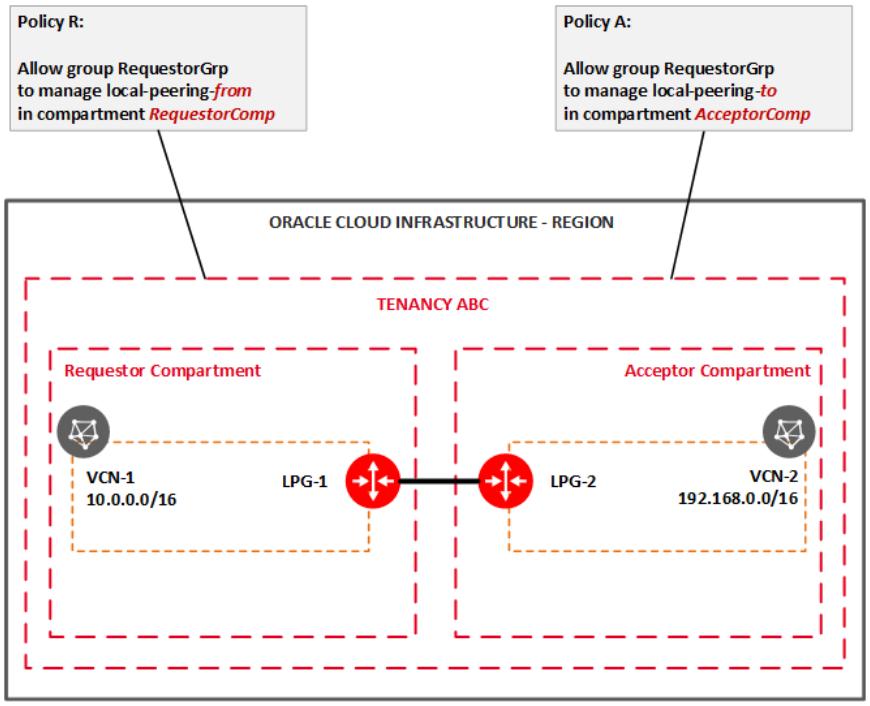
  Allow group RequestorGrp to inspect vcns in compartment AcceptorComp
  Allow group RequestorGrp to inspect local-peering-gateways in compartment AcceptorComp
  ```

The first statement in the policy lets the requestor connect to any LPG in the acceptor's compartment (AcceptorComp). This statement reflects the required agreement from the acceptor for the peering to be established. Policy A can be attached to either the tenancy (root compartment) or to AcceptorComp.



Tip

The second and third statements in Policy A let the requestor list the VCNs and LPGs in AcceptorComp. The statements are required for the requestor to use the Console UI to select from a list of VCNs and LPGs in AcceptorComp and establish the connection. The following diagram focuses only on the first statement, which is the critical one that permits the connection.



Both Policy R and Policy A give RequestorGrp access. However, Policy R has a resource-type called local-peering-*from*, and Policy A has a resource-type called local-peering-*to*. Together, these policies let someone in RequestorGrp establish the connection *from* an LPG in the requestor's compartment *to* an LPG in the acceptor's compartment. The API call to create the connection specifies which two LPGs.



Tip

The permission granted by Policy R might already be in place if the requestor has permission in another policy to manage all of the Networking components in RequesterComp. For example, there might be a general Network Admin policy like this:

```
Allow group NetworkAdmin to manage virtual-network-family in
compartment RequestorComp
```

If the requestor is in the NetworkAdmin group, then they already have the required permissions covered in Policy R (the virtual-network-family includes LPGs). And further, if the policy is instead written to cover the *entire tenancy* instead of only compartment RequestorComp, then the requestor already has all the required permissions in both compartments to establish the connection. In that case, policy A is not required.

Task C: Set up the IAM policies (VCNs in different tenancies)

In this version of task C, the VCNs are in different tenancies (in other words, it's a *cross-tenancy* peering). If the VCNs are in the same tenancy, instead see Task C: Set up the IAM policies (VCNs in same tenancy).

Both the requestor and acceptor must ensure that the right policies are in place:

- **Policy R (implemented by the requestor):**

```
Define tenancy Acceptor as <acceptor_tenancy_OCID>

Allow group RequestorGrp to manage local-peering-from in compartment RequestorComp
```

```
Endorse group RequestorGrp to manage local-peering-to in tenancy Acceptor

Endorse group RequestorGrp to associate local-peering-gateways in compartment RequestorComp
   with local-peering-gateways in tenancy Acceptor
```

The requestor is in an IAM group called RequestorGrp. This policy lets anyone in that group initiate a connection from any LPG in the requestor's compartment (RequestorComp).

The first statement is a "define" statement that assigns a friendly label to the acceptor's tenancy OCID. The statement happens to use "Acceptor" as the label, but it could be a value of the requestor's choice. **All "define" statements in a policy must be the first ones (at the top).**

The second statement lets the RequestorGrp establish a connection from an LPG in the requestor's compartment.

The third and fourth statements are special ones required because the LPGs are in different tenancies. They let the RequestorGrp connect an LPG in the requestor's tenancy to an LPG in the acceptor's tenancy.

If the desire is to give the RequestorGrp permission to connect to an LPG *in any tenancy*, the policy would instead look like this:

```
Allow group RequestorGrp to manage local-peering-from in compartment RequestorComp

Endorse group RequestorGrp to manage local-peering-to in any-tenancy

Endorse group RequestorGrp to associate local-peering-gateways in compartment RequestorComp
    with local-peering-gateways in any-tenancy
```

Regardless, Policy R must be attached to the requestor's tenancy (root compartment), and not the requestor's compartment. Policies that enable cross-tenancy access must be attached to the tenancy. For more information about attachment of policies, see Policy Attachment.

- **Policy A (implemented by the acceptor):**

```
Define tenancy Requestor as <requestor_tenancy_OCID>

Define group RequestorGrp as <RequestorGrp_OCID>
```

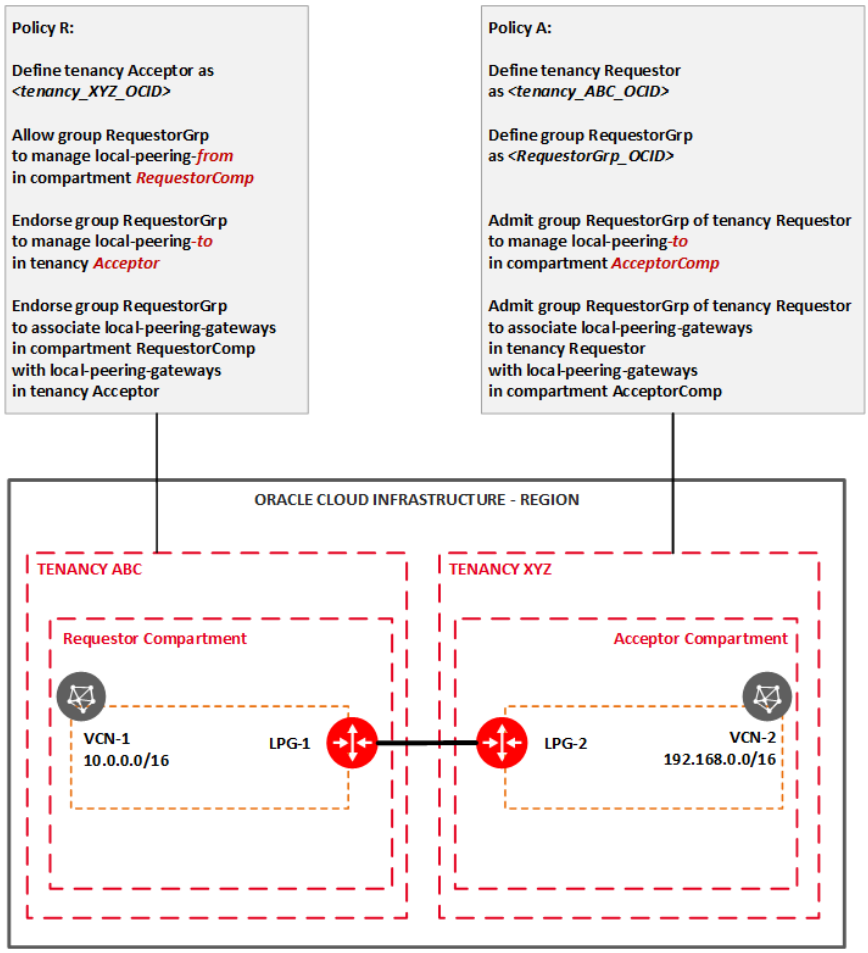
```
Admit group RequestorGrp of tenancy Requestor to manage local-peering-to in compartment
AcceptorComp

Admit group RequestorGrp of tenancy Requestor to associate local-peering-gateways in tenancy
Requestor
    with local-peering-gateways in compartment AcceptorComp
```

Similar to the requestor's policy, this policy first uses "define" statements to assign friendly labels to the requestor's tenancy OCID and the RequestorGrp OCID. As mentioned earlier, the acceptor could use other values for those labels if desired.

The third and fourth statements let the RequestorGrp connect to an LPG in the acceptor's compartment (AcceptorComp). **These statements reflect the critical agreement required for the peering to be established.** The word `Admit` indicates that the access applies to a group *outside the tenancy* where the policy resides.

Policy A must be attached to the acceptor's tenancy (root compartment), and not the acceptor's compartment.



Task D: Establish the connection

The requestor must perform this task.

Prerequisite: The requestor must have the OCID of the acceptor's LPG.



Tip

If you're using the Console and the peering is between two VCNs in the same tenancy: Instead of specifying the acceptor's LPG OCID, you can instead pick the acceptor's VCN and LPG from lists of resources in the tenancy. However, you need to know both the name and compartment for the acceptor's VCN and LPG instead of the LPG's OCID. For reference, see Task B: Share information.

1. In the Console, view the details for the requestor LPG that you want to connect to the acceptor LPG.
2. For the requestor LPG, click the Actions icon (three dots), and then click **Establish Peering Connection**.
3. Specify which LPG you want to peer with:
   - If the VCNs are in different tenancies: Select **Enter Local Peering Gateway OCID**, and enter the acceptor LPG's OCID.
   - If the VCNs are in the same tenancy: Do one of the following:
     - Select **Enter Local Peering Gateway OCID**, and enter the acceptor LPG's OCID.
     - Select **Browse Below**, and then select the acceptor's VCN and LPG from the lists provided. Remember that the VCN and LPG each might be in a different compartment than the one you're currently working in.
4. Click **Establish Peering Connection**.

The connection is established and the LPG's state changes to PEERED.

At this point, the details of each LPG update to show the **Peer VCN CIDR Block** for the other VCN. This is the CIDR of the other VCN across the peering from the LPG. Each administrator can use this information to update the route tables and security lists for their own VCN.

Task E: Configure the route tables

As mentioned earlier, each administrator can do this task before or after the connection is established.

Prerequisite: Each administrator must have the CIDR block or specific subnets for the other VCN. If the connection is already established, look at the **Peer VCN CIDR Block** for your LPG in the Console. Otherwise, get the information from the other administrator by email or other method.

For your own VCN:

1. Determine which subnets in your VCN need to communicate with the other VCN.
2. Update the route table for each of those subnets to include a new rule that directs traffic destined for the other VCN's CIDR to your LPG:
    a. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
    b. Click the VCN you're interested in.
    c. Under **Resources**, click **Route Tables**.
    d. Click the route table you're interested in.
    e. Click **Add Route Rule** and enter the following:
        - **Target Type:** Local Peering Gateway.
        - **Destination CIDR Block:** The other VCN's CIDR block. If you want, you can specify a subnet or particular subset of the peered VCN's CIDR.
        - **Target Compartment:** The compartment where the LPG is located, if not the current compartment.
        - **Target:** The LPG.
    f. Click **Add Route Rule**.

Any subnet traffic with a destination that matches the rule is routed to your LPG. For more information about setting up route rules, see Route Tables.

Later, if you no longer need the peering and want to delete your LPG, you must first delete all the route rules in your VCN that specify the LPG as the target.



Tip

Without the required routing, traffic doesn't flow between the peered LPGs. If a situation occurs where you need to temporarily stop the peering, you can simply remove the route rules that enable traffic. You don't need to delete the LPGs.

Task F: Configure the security lists

As mentioned earlier, each administrator can do this task before or after the connection is established.

Prerequisite: Each administrator must have the CIDR block or specific subnets for the other VCN. In general, you should use the same CIDR block you used in the route table rule in Task E: Configure the route tables.

What rules should you add?

- Ingress rules for the types of traffic you want to allow from the other VCN, specifically from the VCN's CIDR or specific subnets.
- Egress rule to allow outgoing traffic from your VCN to the other VCN. If the subnet already has a broad egress rule for all types of protocols to all destinations (0.0.0.0/0), then you don't need to add a special one for the other VCN.

For your own VCN:

1. Determine which subnets in your VCN need to communicate with the other VCN.
2. Update the security list for each of those subnets to include rules to allow the desired egress or ingress traffic specifically with the CIDR block or subnet of the other VCN:

a. In the Console, while viewing the VCN you're interested in, click **Security Lists**.
b. Click the security list you're interested in.
c. Under **Resources**, click either **Ingress Rules** or **Egress Rules** depending on the type of rule you want to work with.
d. If you want to add a new rule, click **Add Ingress Rule** (or **Add Egress Rule**).

Example

Let's say you want to add a stateful rule that enables ingress HTTPS (port 443) traffic from the other VCN's CIDR. Here are the basic steps you take when adding a rule:

i. Leave the **Stateless** check box unselected.
ii. **Source Type:** Leave as CIDR.
iii. **Source CIDR:** Enter the same CIDR block that the route rules use (see Task E: Configure the route tables).
iv. **IP Protocol:** Leave as TCP.
v. **Source Port Range:** Leave as All.
vi. **Destination Port Range**: Enter 443.

e. If you want to delete an existing rule, click the Actions icon (three dots), and then click **Remove**.
f. If you wanted to edit an existing rule, click the Actions icon (three dots), and then click **Edit**.

For more information about setting up security list rules, see Security Lists.

Using the Console

To create a local peering gateway

See the instructions in Task A: Create the LPGs.

To associate a route table with an existing local peering gateway

This task is related to an advanced routing scenario called transit routing.

An LPG can exist without a route table associated with it. However, after you associate a route table with an LPG, there must always be a route table associated with it. But, you can associate a *different* route table. You can also edit the table's rules, or delete some or all of the rules.

Prerequisite: The route table must exist and belong to the VCN that the LPG belongs to.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Under **Resources**, click **Local Peering Gateways**.
4. For the LPG you're interested in, click the Actions icon (•••), and then click either:
   - **Associate With Route Table:** If the LPG has no route table associated with it yet.
   - **Replace Route Table Association:** If you're changing which route table is associated with the LPG.
5. Select the compartment where the route table resides, and select the route table itself.
6. Click **Associate**.

To delete a local peering gateway

Prerequisite: First delete all the route rules in your VCN that specify the LPG as the target.

Deleting those rules stops the routing in your VCN to the LPG. However, the LPG's state may still be PEERED if the other LPG still exists. Whenever an LPG is deleted, the LPG at the other side of the peering changes to the REVOKED state.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Click **Local Peering Gateways**.
4. For the LPG you want to delete, click the Actions icon (three dots), and then click **Terminate**.
5. Confirm when prompted.



Note

After deleting an LPG (and thus terminating a peering), it's recommended you review your security lists to remove any rules that enabled traffic with the other VCN.

To manage tags for a local peering gateway

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the VCN you're interested in.
3. Under **Resources**, click **Local Peering Gateways**.
4. Click the Actions icon (three dots) for the local peering gateway, and then click **View Tags**. From there you can view the existing tags, edit them, and apply new ones.

For more information, see Resource Tags.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

To manage your LPGs and create connections, use these operations:

- ListLocalPeeringGateways
- CreateLocalPeeringGateway
- GetLocalPeeringGateway
- UpdateLocalPeeringGateway
- DeleteLocalPeeringGateway
- ConnectLocalPeeringGateways

Remote VCN Peering (Across Regions)

This topic is about *remote VCN peering.* In this case, *remote* means that the VCNs reside *in different regions.* If the VCNs you want to connect are in the same region, see Local VCN Peering (Within Region).



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Overview of Remote VCN Peering

Remote VCN peering is the process of connecting two VCNs in different regions (but the same tenancy). The peering allows the VCNs' resources to communicate using private IP addresses

without routing the traffic over the internet or through your on-premises network. Without peering, a given VCN would need an internet gateway and public IP addresses for the instances that need to communicate with another VCN in a different region.

Summary of Networking Components for Remote Peering

At a high level, the Networking service components required for a remote peering include:

- Two VCNs with non-overlapping CIDRs, in different regions that support remote peering. The VCNs must be in the same tenancy.

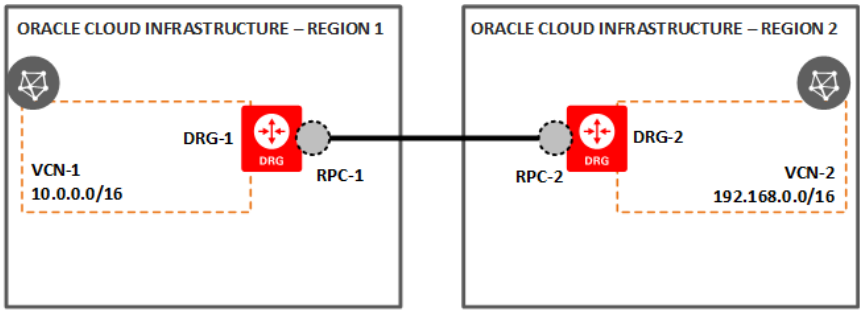
  Note

  All VCN CIDRs Must Not Overlap

  The two VCNs in the peering relationship must not have overlapping CIDRs. Also, if a particular VCN has multiple peering relationships, those other VCNs must not have overlapping CIDRs with each other. For example, if VCN-1 is peered with VCN-2 and also VCN-3, then VCN-2 and VCN-3 must not have overlapping CIDRs.

- A dynamic routing gateway (DRG) attached to each VCN in the peering relationship. Your VCN already has a DRG if you're using an IPSec VPN or an Oracle Cloud Infrastructure FastConnect private virtual circuit.
- A *remote peering connection (RPC)* on each DRG in the peering relationship.
- A *connection* between those two RPCs.
- Supporting route rules to enable traffic to flow over the connection, and only to/from select subnets in the respective VCNs (if desired).
- Supporting security list rules to control the types of traffic allowed to/from the instances in the subnets that need to communicate with the other VCN.

The following diagram illustrates the components.



With supporting route tables and security lists in each VCN to enable traffic

> **Note**
>
> A given VCN can use the connected RPCs to reach only VNICs in the other VCN, and not destinations outside of the VCNs (such as the internet or your on-premises network). For example, if VCN-1 in the preceding diagram were to have an internet gateway, the instances in VCN-2 could NOT use it to send traffic to endpoints on the internet. However, be aware that VCN-2 could *receive* traffic from the internet via VCN-1. For more information, see Important Implications of Peering.

Explicit Agreement Required from Both Sides

Peering involves two VCNs in the same tenancy that might be administered by the same party or two different ones. The two parties might both be in your company but in different departments.

Peering between two VCNs requires explicit agreement from both parties in the form of Oracle Cloud Infrastructure Identity and Access Management policies that each party implements for their own VCN's compartment.

Important Remote Peering Concepts

The following concepts help you understand the basics of VCN peering and how to establish a remote peering.

PEERING

A *peering* is a single peering relationship between two VCNs. Example: If VCN-1 peers with two other VCNs, then there are two peerings. The *remote* part of *remote peering* indicates that the VCNs are in different regions. For remote peering, the VCNs must be in the same tenancy.

VCN ADMINISTRATORS

In general, VCN peering can occur only if both of the VCN administrators agree to it. In practice, this means that the two administrators must:

- Share some basic information with each other.
- Coordinate to set up the required Oracle Cloud Infrastructure Identity and Access Management policies to enable the peering.
- Configure their VCNs for the peering.

Depending on the situation, a single administrator might be responsible for both VCNs and the related policies. The VCNs must be in the same tenancy.

For more information about the required policies and VCN configuration, see Setting Up a Remote Peering.

ACCEPTOR AND REQUESTOR

To implement the IAM policies required for peering, the two VCN administrators must designate one administrator as the *requestor* and the other as the *acceptor*. The requestor must be the one to initiate the request to connect the two RPCs. In turn, the acceptor must create a particular IAM policy that gives the requestor permission to connect to RPCs in the acceptor's compartment. Without that policy, the requestor's request to connect fails.

REGION SUBSCRIPTION

To peer with a VCN in another region, your tenancy must first be subscribed to that region. For information about subscribing, see Managing Regions.

REMOTE PEERING CONNECTION (RPC)

A *remote peering connection (RPC)* is a component you create on the DRG attached to your VCN. The RPC's job is to act as a connection point for a remotely peered VCN. As part of configuring the VCNs, each administrator must create an RPC for the DRG on their VCN. A given DRG must have a separate RPC for each remote peering it establishes for the VCN (maximum 10 RPCs per tenancy). To continue with the previous example: the DRG on VCN-1 would have two RPCs to peer with two other VCNs. In the API, a RemotePeeringConnection is an object that contains information about the peering. You can't reuse an RPC to later establish another peering with it.

CONNECTION BETWEEN TWO RPCS

When the requestor initiates the request to peer (in the Console or API), they're effectively asking to *connect the two RPCs*. This means the requestor must have information to identify each RPC (such as the RPC's region and OCID).

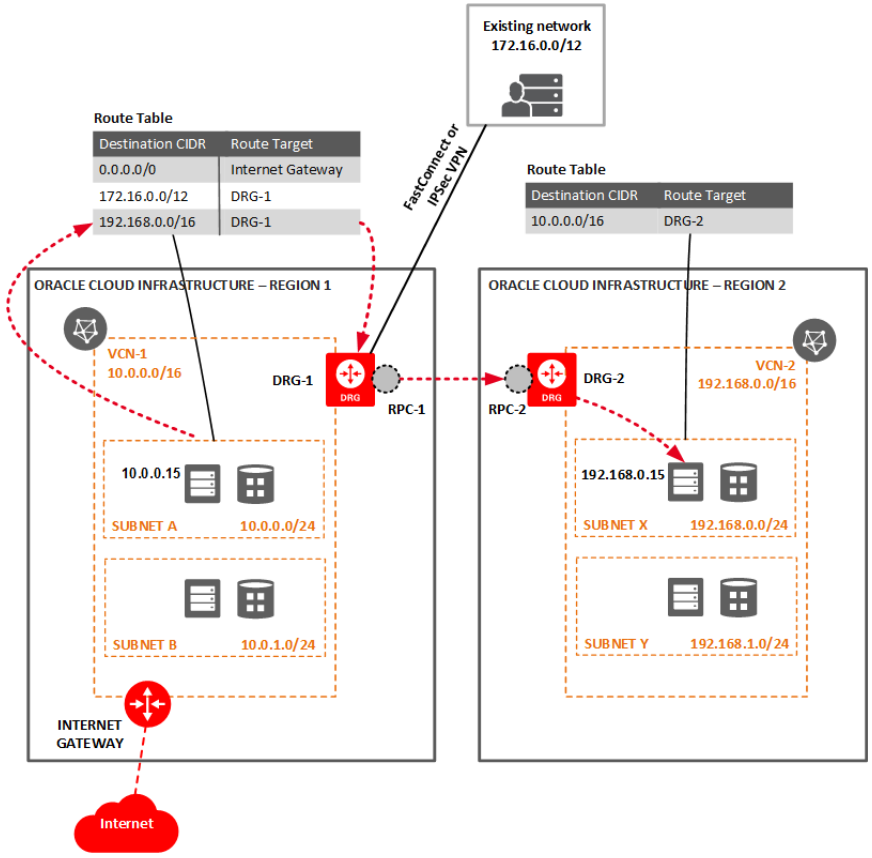
Either VCN administrator can terminate a peering by deleting their RPC. In that case, the other RPC's status switches to REVOKED. The administrator could instead render the connection non-functional by removing the route rules or security lists that enable traffic to flow across the connection (see the next sections).

ROUTING TO THE DRG

As part of configuring the VCNs, each administrator must update the VCN's routing to enable traffic to flow between the VCNs. For each subnet that needs to communicate with the other VCN, you update the subnet's route table. The route rule specifies the destination traffic's CIDR and your DRG as the target. Your DRG routes traffic that matches that rule to the other DRG, which in turn routes the traffic to the next hop in the other VCN.

In the following diagram, VCN-1 and VCN-2 are peered. Traffic from an instance in Subnet A (10.0.0.15) that is destined for an instance in VCN-2 (192.168.0.15) is routed to DRG-1

based on the rule in Subnet A's route table. From there the traffic is routed through the RPCs to DRG-2, and then from there, on to the destination in Subnet X.



Note

As mentioned earlier, a given VCN can use the connected RPCs to reach only VNICs in the other



VCN, and not destinations outside of the VCNs (such as the internet or your on-premises network). For example, in the preceding diagram, VCN-2 cannot use the internet gateway attached to VCN-1.

SECURITY LIST RULES

Each subnet in a VCN has one or more security lists that control traffic in and out of the subnet's VNICs at the packet level. You can use security lists to control the type of traffic allowed with the other VCN. As part of configuring the VCNs, each administrator must determine which subnets in their own VCN need to communicate with VNICs in the other VCN and update their subnet's security lists accordingly.

Important Implications of Peering

If you haven't yet, read Important Implications of Peering to understand important access control, security, and performance implications for peered VCNs.

Setting Up a Remote Peering

This section covers the general process for setting up a peering between two VCNs in different regions.



Important

The following procedure assumes that:

- Your tenancy is subscribed to the other VCN's region. If it's not, see Managing Regions.
- You already have a DRG attached to your VCN. If you don't, see Dynamic Routing Gateways (DRGs).

A. **Create the RPCs:** Each VCN administrator creates an RPC for their own VCN's DRG.
B. **Share information:** The administrators share the basic required information.
C. **Set up the required IAM policies for the connection:** The administrators set up IAM policies to enable the connection to be established.
D. **Establish the connection:** The requestor connects the two RPCs.
E. **Update route tables:** Each administrator updates their VCN's route tables to enable traffic between the peered VCNs as desired.
F. **Update security lists:** Each administrator updates their VCN's security lists to enable traffic between the peered VCNs as desired.

If desired, the administrators can perform tasks E and F *before* establishing the connection. Each administrator needs to know the CIDR block or specific subnets from the other's VCN and share that in task B.

Task A: Create the RPCs

Each administrator creates an RPC for their own VCN's DRG. "You" in the following procedure means an administrator (either the acceptor or requestor).



Note

Required IAM Policy to Create RPCs

If the administrators already have broad network administrator permissions (see Let network admins manage a cloud network), then they have permission to create, update, and delete RPCs. Otherwise, here's an example policy giving the necessary permissions to a group called RPCAdmins. The second statement is required because creating an RPC affects the DRG it belongs to, so the administrator must have permission to manage DRGs.

```
Allow group RPCAdmins to manage remote-peering-connections in
tenancy
Allow group RPCAdmins to manage drgs in tenancy
```

1. In the Console, confirm you're viewing the compartment that contains the DRG that you want to add the RPC to. For information about compartments and access control, see Access Control.
2. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Dynamic Routing Gateways**.
3. Click the DRG you're interested in.
4. Under **Resources**, click **Remote Peering Connections**.
5. Click **Create Remote Peering Connection**.
6. Enter the following:
   - **Name:** A friendly name for the RPC. It doesn't have to be unique, and it cannot be changed later in the Console (but you can change it with the API). Avoid entering confidential information.

- **Create in compartment**: The compartment where you want to create the RPC, if different from the compartment you're currently working in.

7. Click **Create Remote Peering Connection**.
   The RPC is then created and displayed on the **Remote Peering Connections** page in the compartment you chose.
8. If you're the acceptor, record the RPC's region and OCID to later give to the requestor.

Task B: Share information

- If you're the acceptor, give this information to the requestor (for example, by email or other out-of-band method):
  - The region your VCN is in (the requestor's tenancy must be subscribed to this region).
  - Your RPC's OCID.
  - The CIDR blocks for subnets in your VCN that should be available to the other VCN. The requestor needs this information when setting up routing for the requestor VCN.
- If you're the requestor, give this information to the acceptor:
  - The region your VCN is in (the acceptor's tenancy must be subscribed to this region).
  - The name of the IAM group that should be granted permission to create a connection in the acceptor's compartment (in the example in the next task, the group is RequestorGrp).
  - The CIDR blocks for subnets in your VCN that should be available to the other VCN. The acceptor needs this information when setting up routing for the acceptor VCN.

Task C: Set up the IAM policies

Both the requestor and acceptor must ensure the right policies are in place. These consist of:

- **Policy R (implemented by the requestor):**

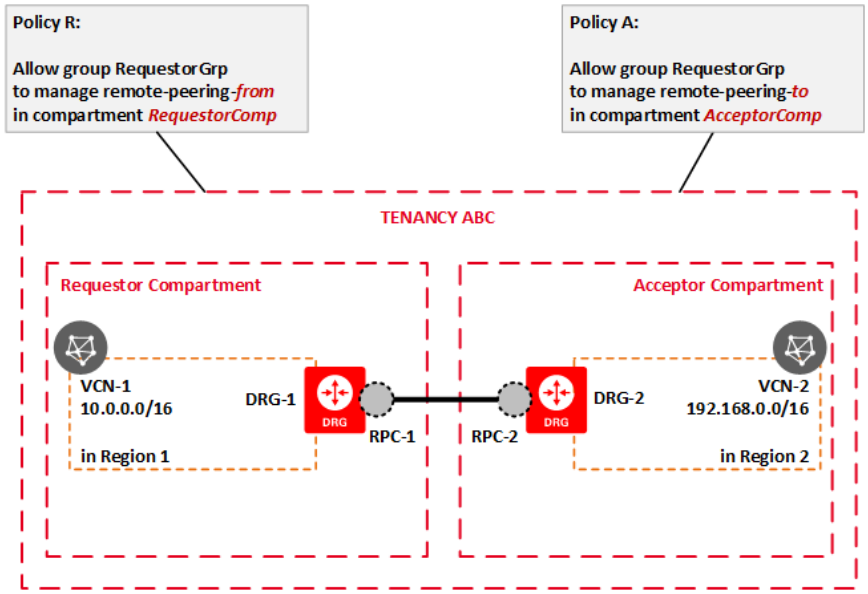
```
Allow group RequestorGrp to manage remote-peering-from in compartment RequestorComp
```

  The requestor is in an IAM group called RequestorGrp. This policy lets anyone in the group initiate a connection from any RPC in the requestor's compartment (RequestorComp). Policy R can be attached to either the tenancy (root compartment) or to RequestorComp. For information about why you would attach it to one versus the other, see Policy Attachment.

- **Policy A (implemented by the acceptor):**

```
Allow group RequestorGrp to manage remote-peering-to in compartment AcceptorComp
```

  This policy lets the requestor connect to any RPC in the acceptor's compartment (AcceptorComp). This statement reflects the required agreement from the acceptor for the peering to be established. Policy A can be attached to either the tenancy (root compartment) or to AcceptorComp.



Both Policy R and Policy A give RequestorGrp access. However, Policy R has a resource-type called remote-peering-*from*, and Policy A has a resource-type called remote-peering-*to*. Together, these policies let someone in RequestorGrp establish the connection *from* an RPC in the requestor's compartment *to* an RPC in the acceptor's compartment. The API call to actually create the connection specifies which two RPCs.



Tip

The permission granted by Policy R might already be in place if the requestor has permission in another policy to manage all of the Networking components in RequesterComp. For example, there might be a general Network Admin policy like this: `Allow group NetworkAdmin to manage virtual-network-family in compartment RequestorComp`. If the requestor is in the NetworkAdmin group, then they already have the required permissions covered in Policy R (the virtual-network-family includes RPCs). And further, if the policy is instead written to cover the *entire tenancy* (`Allow group NetworkAdmin to manage virtual-network-family in tenancy`), then the requestor already has all the required permissions in both compartments to establish the connection. In that case, policy A is not required.

Task D: Establish the connection

The requestor must perform this task.

Prerequisite: The requestor must have:

- The region the acceptor's VCN is in (the requestor's tenancy must be subscribed to the region).
- The OCID of the acceptor's RPC.

1. In the Console, view the details for the requestor RPC that you want to connect to the acceptor RPC.
2. Click **Establish Connection**.
3. Enter the following:
    - **Region:** The region that contains the acceptor's VCN. The drop-down list includes only those regions that both support remote VCN peering and your tenancy is subscribed to.
    - **Remote Peering Connection ID:** The OCID of the acceptor's RPC.
4. Click **Establish Connection**.

The connection is established and the RPC's state changes to PEERED.

Task E: Configure the route tables

As mentioned earlier, each administrator can do this task before or after the connection is established.

Prerequisite: Each administrator must have the CIDR block or specific subnets for the other VCN.

For your own VCN:

1. Determine which subnets in your VCN need to communicate with the other VCN.
2. Update the route table for each of those subnets to include a new rule that directs traffic destined for the other VCN to your DRG:
    a. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
    b. Click the VCN you're interested in.

c. Under **Resources**, click **Route Tables**.

d. Click the route table you're interested in.

e. Click **Add Route Rule** and enter the following:

- **Target Type:** Dynamic Routing Gateway. The VCN's attached DRG is automatically selected as the target, and you don't have to specify the target yourself.
- **Destination CIDR Block:** The other VCN's CIDR block. If you want, you can specify a subnet or particular subset of the peered VCN's CIDR.

f. Click **Add Route Rule**.

Any subnet traffic with a destination that matches the rule is routed to your DRG. For more information about setting up route rules, see Route Tables.



Tip

Without the required routing, traffic doesn't flow between the peered DRGs. If a situation occurs where you need to temporarily stop the peering, you can simply remove the route rules that enable traffic. You don't need to delete the RPCs.

Task F: Configure the security lists

As mentioned earlier, each administrator can do this task before or after the connection is established.

Prerequisite: Each administrator must have the CIDR block or specific subnets for the other VCN. In general, you should use the same CIDR block you used in the route table rule in Task E: Configure the route tables.

What rules should you add?

- Ingress rules for the types of traffic you want to allow from the other VCN, specifically from the VCN's CIDR or specific subnets.
- Egress rule to allow outgoing traffic from your VCN to the other VCN. If the subnet already has a broad egress rule for all types of protocols to all destinations (0.0.0.0/0), then you don't need to add a special one for the other VCN.

For your own VCN:

1. Determine which subnets in your VCN need to communicate with the other VCN.
2. Update the security list for each of those subnets to include rules to allow the desired egress or ingress traffic specifically with the CIDR block or subnet of the other VCN:
    a. In the Console, while viewing the VCN you're interested in, click **Security Lists**.
    b. Click the security list you're interested in.
    c. Under **Resources**, click either **Ingress Rules** or **Egress Rules** depending on the type of rule you want to work with.
    d. If you want to add a new rule, click **Add Ingress Rule** (or **Add Egress Rule**).

    ### Example

    Let's say you want to add a stateful rule that enables ingress HTTPS (port 443) traffic from the other VCN's CIDR. Here are the basic steps you take when adding a rule:

    i. In the **Allow Rules for Ingress** section, click **+Add Rule**.
    ii. Leave the **Stateless** checkbox unchecked.
    iii. **Source Type:** Leave as CIDR.
    iv. **Source CIDR:** Enter the same CIDR block that the route rules use (see Task E: Configure the route tables).
    v. **IP Protocol:** Leave as TCP.

vi. **Source Port Range:** Leave as All.

vii. **Destination Port Range**: Enter 443.

e. If you want to delete an existing rule, click the Actions icon (three dots), and then click **Remove**.

f. If you wanted to edit an existing rule, click the Actions icon (three dots), and then click **Edit**.

For more information about setting up security list rules, see Security Lists.

Using the Console

To create a remote peering connection

See the instructions in Task A: Create the RPCs.

To delete a remote peering connection

Deleting an RPC terminates the peering. The RPC at the other side of the peering changes to the REVOKED state.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Dynamic Routing Gateways**.
2. Click the DRG you're interested in.
3. Under **Resources**, click **Remote Peering Connections**.
4. Click the RPC you're interested in.
5. Click **Terminate**.
6. Confirm when prompted.



Note

After deleting an RPC (and thus terminating a peering), it's recommended you review your route tables and security lists to remove any rules that enabled traffic with the other VCN.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

To manage your RPCs and create connections, use these operations:

- ListAllowedPeerRegionsForRemotePeering
- ListRemotePeeringConnections
- CreateRemotePeeringConnection
- GetRemotePeeringConnection
- UpdateRemotePeeringConnection
- DeleteRemotePeeringConnection
- ConnectRemotePeeringConnections

Access to Oracle Cloud Infrastructure Classic

There are two ways to set up a connection between an Oracle Cloud Infrastructure Classic IP network and an Oracle Cloud Infrastructure virtual cloud network (VCN):

- Option 1: Connection over the Oracle network
    - You file a ticket with My Oracle Support and Oracle provisions a connection between the IP network's private gateway and the VCN's attached dynamic routing gateway (DRG). The connection runs over Oracle's network and not the internet.
    - The two environments must be in the same geographical area (Ashburn or London), and the connection is available only between these specific regions:
        - Between Oracle Cloud Infrastructure us-ashburn-1 region and Oracle Cloud Infrastructure Classic uscom-east-1 region
        - Between Oracle Cloud Infrastructure uk-london-1 region and Oracle Cloud Infrastructure Classic gbcom-south-1 region
    - The two environments must belong to the same company. Oracle validates this when setting up the connection.
- Option 2: Connection over an IPSec VPN
    - You set up an IPSec VPN between the IP network's VPN as a Service (VPNaaS) gateway and the VCN's attached DRG. The connection runs over the internet.
    - The two environments do not have to be in the same geographical area or regions.
    - The two environments do not have to belong to the same company.

Connection Over Oracle Network

This topic describes one way to set up a connection between an Oracle Cloud Infrastructure Classic IP network and an Oracle Cloud Infrastructure virtual cloud network (VCN). The connection runs over Oracle's network.

Another option is to connect the two clouds with an IPSec VPN. For more information, see Connection Over IPSec VPN.

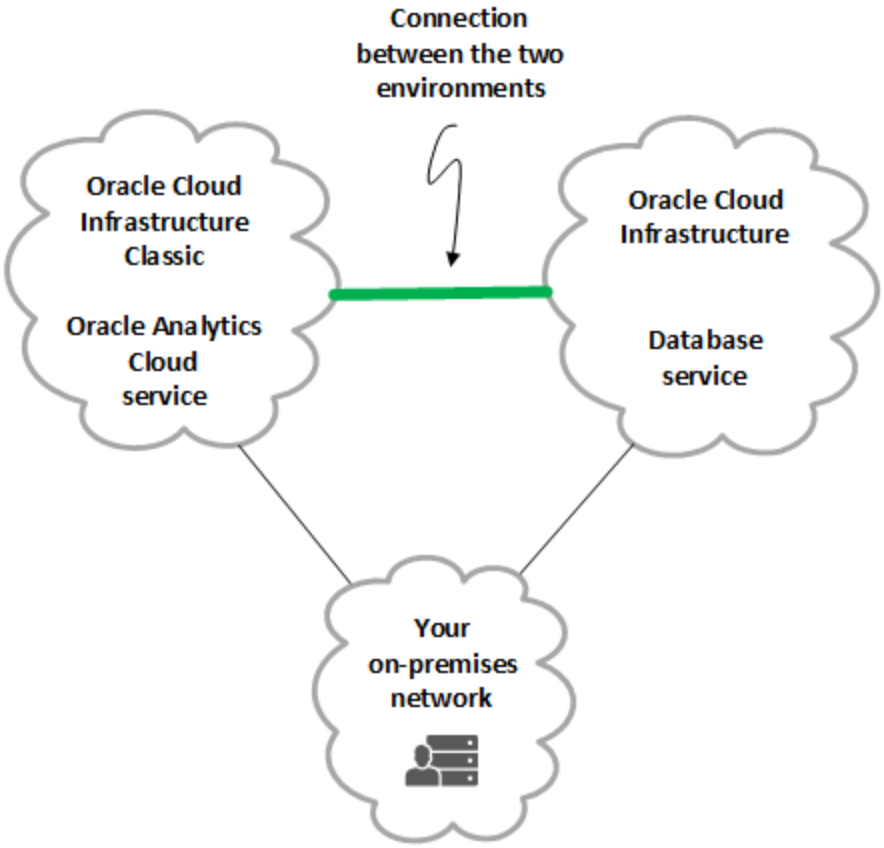
Highlights

- You can run a hybrid workload between your Oracle Cloud Infrastructure Classic and Oracle Cloud Infrastructure environments.
- Oracle connects the IP network's private gateway to the VCN's attached dynamic routing gateway (DRG). The connection runs over the Oracle network. You configure routing and security rules in the environments to enable traffic.
- The two environments must belong to the same company and not have overlapping CIDRs. The cloud resources can communicate over the connection only with private IP addresses.
- The two environments must both be in either the Ashburn area or the London area, and in specific regions listed in the next section. Connectivity to other regions is not supported.
- The connection is free of charge.

Overview

You can request Oracle to provision a connection between your Oracle Cloud Infrastructure environment and your Oracle Cloud Infrastructure Classic environment. The connection facilitates a hybrid deployment with application components that are set up across the two environments. You can also use the connection to migrate workloads from Oracle Cloud Infrastructure Classic to Oracle Cloud Infrastructure. Compared to an IPSec VPN: the resources in the two environments have a more reliable and consistent network connection, with better throughput, because the traffic uses Oracle's internal links. Compared to FastConnect: you don't incur the additional cost and operational overhead of working with a FastConnect partner.

The following diagram shows an example of a hybrid deployment. Oracle Analytics Cloud is running in an Oracle Cloud Infrastructure Classic IP network and accessing the Database service in Oracle Cloud Infrastructure over the connection.



Here are other important details to know:

- The connection is supported only between these regions:
  - Oracle Cloud Infrastructure us-ashburn-1 region and Oracle Cloud Infrastructure Classic uscom-east-1 region
  - Oracle Cloud Infrastructure uk-london-1 region and Oracle Cloud Infrastructure Classic gbcom-south-1 region
- The connection enables communication that uses private IP addresses only.

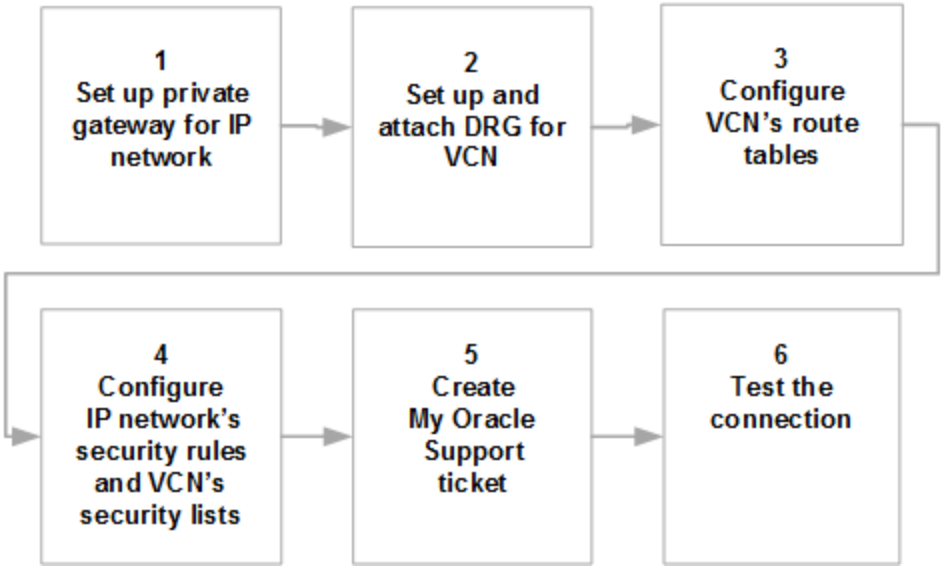
- The CIDR blocks of the IP network and VCN subnets that need to communicate must not overlap.
- The IP network and VCN must belong to the same company. Oracle validates this when setting up the connection.
- This connection enables communication only between resources in the Oracle Cloud Infrastructure Classic IP network and Oracle Cloud Infrastructure VCN. It does not enable traffic between your on-premises network through the IP network to the VCN, or from your on-premises network through the VCN to the IP network.
- The connection also does not enable traffic to flow from the IP network through the connected VCN to a peered VCN in the same Oracle Cloud Infrastructure region, or a different region.

The following table lists the comparable networking components required on each side of the connection.

Component	Oracle Cloud Infrastructure Classic	Oracle Cloud Infrastructure
Cloud network	IP network	VCN
Gateway	private gateway	dynamic routing gateway (DRG)
Routing	routes	route tables with route rules
Security rules	security rules	security lists

Connecting Your IP Network and VCN

The following flow chart shows the overall process of connecting your IP network and VCN.



Prerequisites:

You must already have:

- An Oracle Cloud Infrastructure Classic IP network.
- An Oracle Cloud Infrastructure VCN with subnets.

Task 1: Set up a private gateway for your IP network

If you do not already have a private gateway for your IP network, create one.

Task 2: Set up a dynamic routing gateway (DRG) for your VCN

If you do not already have a DRG attached to your VCN, create a DRG and attach it:

- To create a dynamic routing gateway
- To attach a dynamic routing gateway to a VCN

Task 3: Configure route tables

For the IP network

When you create the private gateway and attach an IP network to it, traffic from cloud resources in the IP network uses the private gateway as the next hop. You do not need to update any routes for the IP network.

For the VCN

You must add a route rule that directs traffic from the VCN's subnets to the DRG:

1. Determine which subnets in your VCN need to communicate with the IP network.
2. Update the route table for each of those subnets to include a new rule that directs traffic destined for the IP network's CIDR to your DRG:
    a. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
    b. Click the VCN you're interested in.
    c. Under **Resources**, click **Route Tables**.
    d. Click the route table you're interested in.
    e. Click **Add Route Rule** and enter the following:
        - **Destination CIDR Block:** The IP network's CIDR block.
        - **Target Type:** Dynamic Routing Gateway. The VCN's attached DRG is automatically selected as the target, and you don't have to specify the

target yourself.

f. Click **Add Route Rule**.

Any subnet traffic with a destination that matches the rule is routed to your DRG. For more information about setting up route rules, see Route Tables.

Later, if you no longer need the connection and want to delete your DRG, you must first delete all the route rules in your VCN that specify the DRG as the target.

Task 4: Configure the security rules and security lists

To ensure traffic flows between the IP network and VCN, make sure the IP network security rules and the VCN's security lists allow the desired traffic.

Here are the types of rules to add:

- Ingress rules for the types of traffic you want to allow into one cloud from the other, specifically from the other cloud's CIDR block.
- Egress rule to allow outgoing traffic from one cloud to the other. If the VCN's subnet already has a broad egress rule for all types of protocols to all destinations (0.0.0.0/0), then you don't need to add a special one for the IP network.

For the IP network

Configure the network security rules for the IP network to allow the desired traffic.

For the VCN

1. Determine which subnets in your VCN need to communicate with the IP network.
2. Update the security list for each of those subnets to include rules to allow the desired egress or ingress traffic specifically with the CIDR block of the IP network:

a. In the Console, while viewing the VCN you're interested in, click **Security Lists**.
b. Click the security list you're interested in.
   Under **Resources**, you can click **Ingress Rules** or **Egress Rules** to switch between the different types of rules.
c. Add one or more rules, each for the specific type of traffic you want to allow.

   Example:

   Let's say you want to add a stateful rule that enables ingress HTTPS (port 443) traffic from the IP network's CIDR. Here are the basic steps you take when adding a rule:

   i. On the **Ingress Rules** page, click **Add Ingress Rule**.
   ii. Leave the **Stateless** check box unselected.
   iii. **Source CIDR:** Enter the same CIDR block that the route rules use (see Task 3: Configure route tables).
   iv. **IP Protocol:** Leave as TCP.
   v. **Source Port Range:** Leave as All.
   vi. **Destination Port Range**: Enter 443.
   vii. Click **Add Ingress Rule**.

For more information about setting up security list rules, see Security Lists.

Task 5: Create a My Oracle Support ticket

To have Oracle set up the connection, create a ticket at My Oracle Support and provide the following information:

- Ticket name: Create IP Network - VCN Connection - *<your_company_name>* - Ashburn
- OCI-C identity domain

- OCI-C private gateway name
- Region
- OCI tenancy OCID
- OCI DRG OCID

For example:

- Ticket name: Create IP Network - VCN Connection - ACME - Ashburn
- OCI-C identity domain: 123456789, uscom-east-1
- OCI-C private gateway name: Compute-acme/jack.jones@example.com/privategateway1
- Region: uscom-east-1 (OCI-C) / us-ashburn-1 (OCI)
- OCI tenancy OCID: ocid1.tenancy.oc1..examplefbpnk5cmdl7gkr6kcakfqmvhvbpcv
- OCI DRG OCID: ocid1.drg.oc1.iad.exampleutg6cmd3fqwqbea7ctadcatm

It can take 3 to 4 business days before your My Oracle Support ticket is complete and the connection is ready to test.

Task 6: Test the connection

After you receive confirmation from your support person that the connection is ready, test the connection. Depending on how you've set up your IP network's security rules and VCN security lists, you should be able to launch an instance in your VCN and access it from an instance in the IP network. Or you should be able to connect from the VCN instance to an instance in the IP network. If you can, your connection is ready to use.

Terminating the Connection

If you want to terminate the connection, file a ticket at My Oracle Support.

Connection Over IPSec VPN

This topic describes one way to set up a connection between an Oracle Cloud Infrastructure Classic IP network and an Oracle Cloud Infrastructure virtual cloud network (VCN). The connection runs over an IPSec VPN.

Another option is to have Oracle set up a connection over the Oracle network. For more information, see Connection Over Oracle Network.

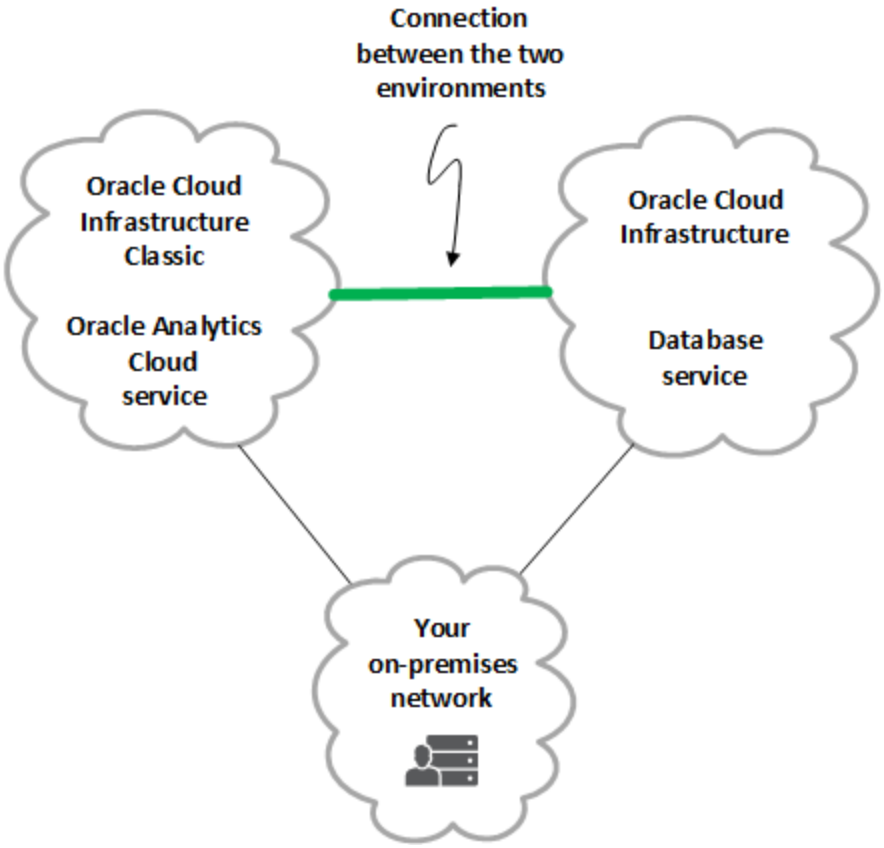
Highlights

- You can run a hybrid workload between your Oracle Cloud Infrastructure Classic and Oracle Cloud Infrastructure environments.
- You set up an IPSec VPN between the IP network's VPN as a Service (VPNaaS) gateway and the VCN's attached dynamic routing gateway (DRG). The connection runs over the internet. You configure routing and security rules in the environments to enable traffic.
- The two environments must not have overlapping CIDRs. The cloud resources can communicate over the connection only with private IP addresses.
- The two environments do not have to be in the same geographical area or region.
- The connection is free of charge.

Overview

You can connect your Oracle Cloud Infrastructure environment and your Oracle Cloud Infrastructure Classic environment with an IPSec VPN. The connection facilitates a hybrid deployment with application components that are set up across the two environments. You can also use the connection to migrate workloads from Oracle Cloud Infrastructure Classic to Oracle Cloud Infrastructure. Compared to using the Oracle network for the connection: you can set up the IPSec VPN yourself in a matter of minutes. Compared to FastConnect: you don't incur the additional cost and operational overhead of working with a FastConnect partner.

The following diagram shows an example of a hybrid deployment. Oracle Analytics Cloud is running in an Oracle Cloud Infrastructure Classic IP network and accessing the Database service in Oracle Cloud Infrastructure over the connection.



Here are other important details to know:

- The connection is supported in any of the Oracle Cloud Infrastructure and Oracle Cloud Infrastructure Classic regions. The two environments do not need to be in the same geographical area.
- The connection enables communication that uses private IP addresses only.
- The CIDR blocks of the IP network and VCN subnets that need to communicate must not overlap.
- This connection enables communication only between resources in the Oracle Cloud Infrastructure Classic IP network and Oracle Cloud Infrastructure VCN. It does not

enable traffic between your on-premises network through the IP network to the VCN, or from your on-premises network through the VCN to the IP network.

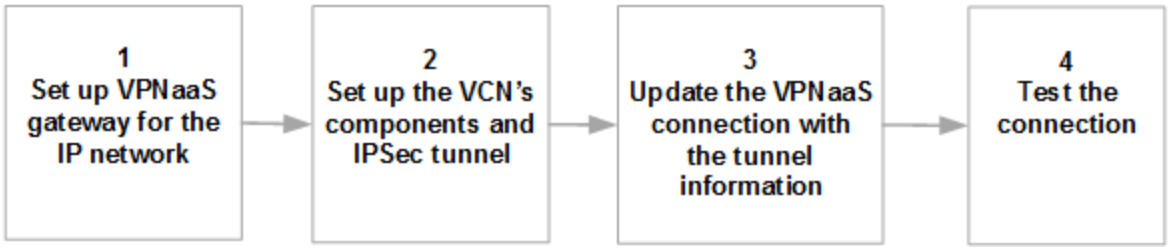
- The connection also does not enable traffic to flow from the IP network through the connected VCN to a peered VCN in the same Oracle Cloud Infrastructure region, or a different region.

The following table lists the comparable networking components required on each side of the connection.

Component	Oracle Cloud Infrastructure Classic	Oracle Cloud Infrastructure
Cloud network	IP network	VCN
Gateway	VPNaaS gateway	dynamic routing gateway (DRG)
Security rules	security rules	security lists

Setting Up the IPSec VPN Between Your IP Network and VCN

The following flow chart shows the overall process of connecting your IP network and VCN with an IPSec VPN.



Prerequisites:

You must already have:

- An Oracle Cloud Infrastructure Classic IP network.
- An Oracle Cloud Infrastructure VCN with subnets.

Task 1: Set up a VPNaaS gateway for your IP network

1. Use these values when setting up the VPNaaS gateway:
    - **IP Network:** The Oracle Cloud Infrastructure Classic IP network you want to connect to your VCN. Note that you can specify only a single IP network.
    - **Customer Gateway:** A placeholder value such as 129.213.240.51. Using this placeholder value lets you move forward in the process. You will change the value later with the Oracle Cloud Infrastructure VPN router's IP address.
    - **Customer Reachable Routes:** The CIDR block for the VCN. Note that you can specify only a single VCN.
    - **Specify Phase 2 ESP Proposal:** Check box selected.
    - **ESP Encryption:** AES 256
    - **ESP Hash:** SHA1
    - **IPSec Lifetime:** 1800
    - **Require Perfect Forward Secrecy:** Check box selected.
2. Record the resulting public IP address of the VPNaaS gateway.

Task 2: Set up the VCN's components and IPSec tunnel

Task 2a: Set up a dynamic routing gateway (DRG) for your VCN

If you do not already have a DRG attached to your VCN, create a DRG and attach it:

- To create a dynamic routing gateway
- To attach a dynamic routing gateway to a VCN

Task 2b: Configure routing to the DRG

You must add a route rule that directs traffic from the VCN's subnets to the DRG. Use the IP network's CIDR block as the destination for the rule.

1. Determine which subnets in your VCN need to communicate with the IP network.
2. Update the route table for each of those subnets to include a new rule that directs traffic destined for the IP network's CIDR to your DRG:
    a. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
    b. Click the VCN you're interested in.
    c. Under **Resources**, click **Route Tables**.
    d. Click the route table you're interested in.
    e. Click **Add Route Rule** and enter the following:
        - **Destination CIDR Block:** The IP network's CIDR block.
        - **Target Type:** Dynamic Routing Gateway. The VCN's attached DRG is automatically selected as the target, and you don't have to specify the target yourself.
    f. Click **Add Route Rule**.

Any subnet traffic with a destination that matches the rule is routed to your DRG. For more information about setting up route rules, see Route Tables.

Later, if you no longer need the connection and want to delete your DRG, you must first delete all the route rules in your VCN that specify the DRG as the target.

Task 2c: Configure the security rules and security lists

To ensure traffic flows between the IP network and VCN, make sure the IP network security rules and the VCN's security lists allow the desired traffic.

Here are the types of rules to add:

- Ingress rules for the types of traffic you want to allow into one cloud from the other, specifically from the other cloud's CIDR block.
- Egress rule to allow outgoing traffic from one cloud to the other. If the VCN's subnet already has a broad egress rule for all types of protocols to all destinations (0.0.0.0/0), then you don't need to add a special one for the IP network.

For the IP network

Configure the network security rules for the IP network to allow the desired traffic.

For the VCN



Important

The VCN's default security list does not allow ICMP echo reply and echo request (ping). You must add rules to your security lists to enable that traffic.

1. Determine which subnets in your VCN need to communicate with the IP network.
2. Update the security list for each of those subnets to include rules to allow the desired egress or ingress traffic specifically with the CIDR block of the IP network:
    a. In the Console, while viewing the VCN you're interested in, click **Security Lists**.
    b. Click the security list you're interested in.
    Under **Resources**, you can click **Ingress Rules** or **Egress Rules** to switch between the different types of rules.
    c. Add one or more rules, each for the specific type of traffic you want to allow.

Example

Let's say you want to add a stateful rule that enables ingress HTTPS (port 443) traffic from the IP network's CIDR. Here are the basic steps you take when adding a rule:

i. On the **Ingress Rules** page, click **Add Ingress Rule**.
ii. Leave the **Stateless** check box unselected.
iii. **Source CIDR:** Enter the same CIDR block that the route rules use (see Task 2b: Configure routing to the DRG).
iv. **IP Protocol:** Leave as TCP.
v. **Source Port Range:** Leave as All.
vi. **Destination Port Range**: Enter 443.
vii. Click **Add Ingress Rule**.

For more information about setting up security list rules, see Security Lists.

Task 2d: Create a CPE object

Create a CPE object. You must provide an IP address. Use the VPNaaS gateway's public IP address.

Task 2e: Create the IPSec connection

From your DRG, create an IPSec connection to the CPE object. You must provide one or more static routes. The values must match the IP network's subnets or aggregate.

The resulting IPSec connection consists of two tunnels. Record the IP address and shared secret for one of those tunnels. In the next task, you will provide those values.

Task 3: Update the VPNaaS connection with the tunnel information

Update the VPNaaS connection. Use these values:

- **Customer Gateway:** The tunnel's IP address from the preceding task.
- **VPNaaS VPN Connection's Pre-shared Key:** The tunnel's shared secret from the preceding task.

After the VPNaaS connection is updated and provisioned, the state of your VCN's IPSec tunnel should change to Available. This might take a few minutes.

Task 4: Test the connection

After the tunnel state changes to Available, test the connection. Depending on how you've set up your IP network's security rules and VCN security lists, you should be able to launch an instance in your VCN and access it from an instance in the IP network. Or you should be able to connect from the VCN instance to an instance in the IP network. If you can, your connection is ready to use.

Terminating the Connection

If you want to terminate the connection, delete the IPSec connection:

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **IPSec Connections**.
   A list of the IPSec connections in the compartment you're viewing is displayed. If you don't see the one you're looking for, verify that you're viewing the correct compartment (select from the list on the left side of the page).
2. Click the IPSec connection you're interested in.
3. Click **Terminate**.
4. Confirm the deletion when prompted.

The IPSec connection will be in the Terminating state for a short period while it's being deleted.

Access to Other Clouds with Libreswan

This topic shows how to connect your Oracle Cloud Infrastructure virtual cloud network (VCN) with another cloud provider by using an IPSec VPN with a Libreswan VM as the customer-premises equipment (CPE).

In the example shown here, the other cloud provider is Amazon Web Services (AWS). The connection is a secure and encrypted site-to-site IPSec VPN between the Oracle and Amazon environments. It enables resources in the two clouds to communicate with each other using their private IP addresses as if they are in the same network segment.

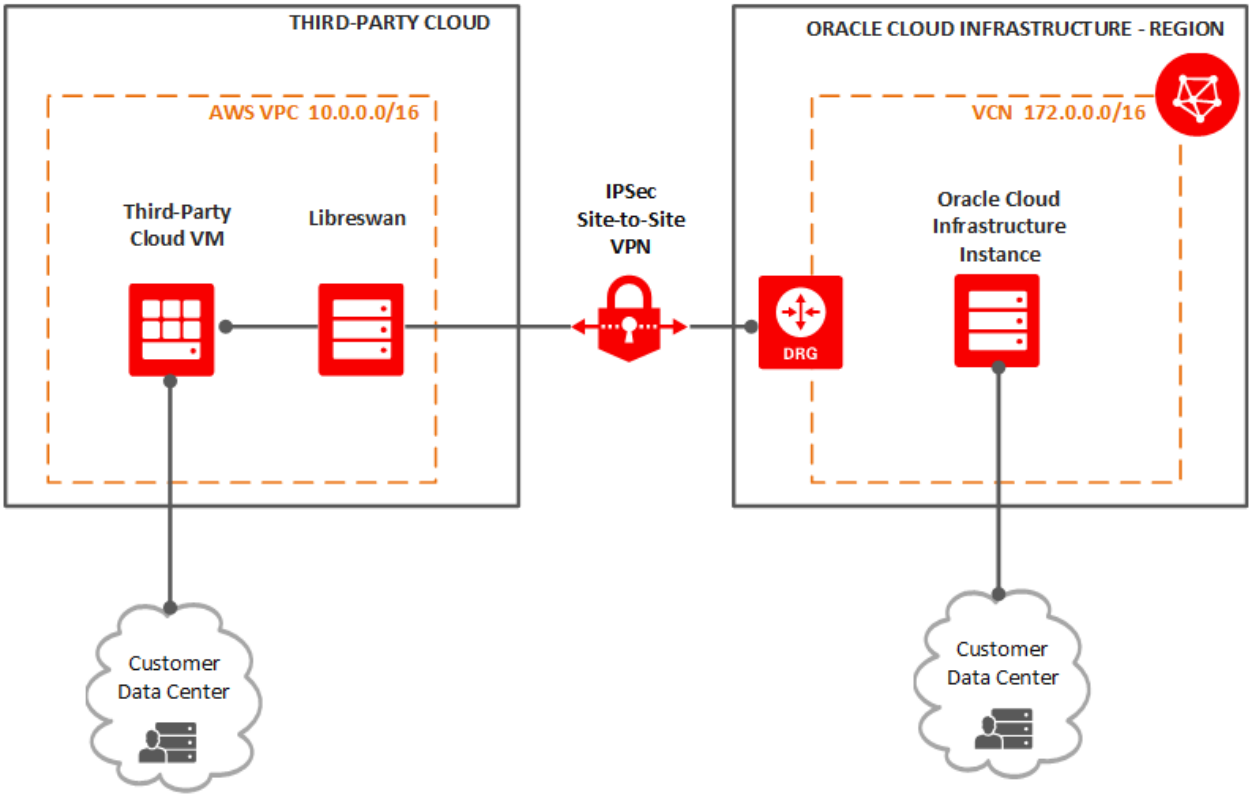
This configuration was validated using Libreswan version 3.23-4.

Route-Based VPNs with Libreswan

Libreswan supports both route-based and policy-based tunnels. The tunnel types can coexist without interfering with each other. The Oracle VPN headends use route-based tunnels. Oracle recommends that you configure Libreswan with the Virtual Tunnel Interface (VTI) configuration syntax.

Architecture

The following diagram shows the general layout of the connection.



Required Personnel and Knowledge

Typically the following types of personnel are involved in setting up an IPSec VPN with Oracle Cloud Infrastructure:

- **Dev Ops team member** (or similar function) who uses the Oracle Cloud Infrastructure Console to set up the cloud components required for the virtual network and IPSec VPN.
- **Network engineer** (or similar function) who configures the on-premises router with information provided by the Dev Ops team member.



Tip

The Dev Ops team member must have the required permission to create and manage the cloud components. If the person is the default administrator for your Oracle Cloud Infrastructure tenancy or a member of the Administrators group, then they have the required permission. For information about restricting access to your networking components, see Access Control.

The personnel should be familiar with the following concepts and definitions:

- The fundamentals of Oracle Cloud Infrastructure
- The basic Networking service components
- General IPSec VPN tunnel functionality

CLOUD RESOURCES

Anything you provision on a cloud platform. For example, with Oracle Cloud Infrastructure, a cloud resource can refer to a VCN, compute instance, user, compartment, load balancer, or any other service component on the platform.

THIRD-PARTY CLOUD PROVIDER

Another cloud provider such as AWS. You must understand that cloud provider's provisioning and configuration process so you can create the resources required to establish an IPSec VPN between the third-party cloud and your Oracle Cloud Infrastructure VCN.

ON-PREMISES

A widely used term in cloud technologies that refers to your traditional data center environments. On-premises can refer to a colocation scenario, a dedicated floor space, a dedicated data center building, or a desktop running under your desk.

ORACLE CLOUD IDENTIFIER (OCID)

A unique identifier assigned to each resource that you provision on Oracle Cloud Infrastructure. The OCID is a long string that Oracle automatically generates. You can't choose the value for an OCID or change a resource's OCID. For more information, see Resource Identifiers.

Information to Gather

The following table lists basic networking values required to set up the connection between the Oracle Cloud Infrastructure VCN and the AWS VPC. In the following table, "User" is you or someone in your company.

Value and Variable Used in the Config File Template	Source	Example Value and Notes
Oracle VCN CIDR block `${VcnCidrBlock}`	User (Oracle)	172.0.0.0/16 The CIDR your company chose for your Oracle Cloud Infrastructure VCN.
AWS VPC CIDR block	User (AWS)	10.0.0.0/16 The CIDR your company chose for your AWS VPC.
AWS Libreswan public IP `${cpePublicIpAddress}`	User (AWS)	AWS VM Public IP - 34.200.255.174 The public IP address for your Libreswan host.
AWS Libreswan local IP `${cpeLocalIP}`	User (AWS)	AWS VM Private IP - 10.1.2.3 The IP address configured directly on your Libreswan host. If your host is not behind 1-1 NAT, this value is the same as the Libreswan public interface (34.220.255.174).

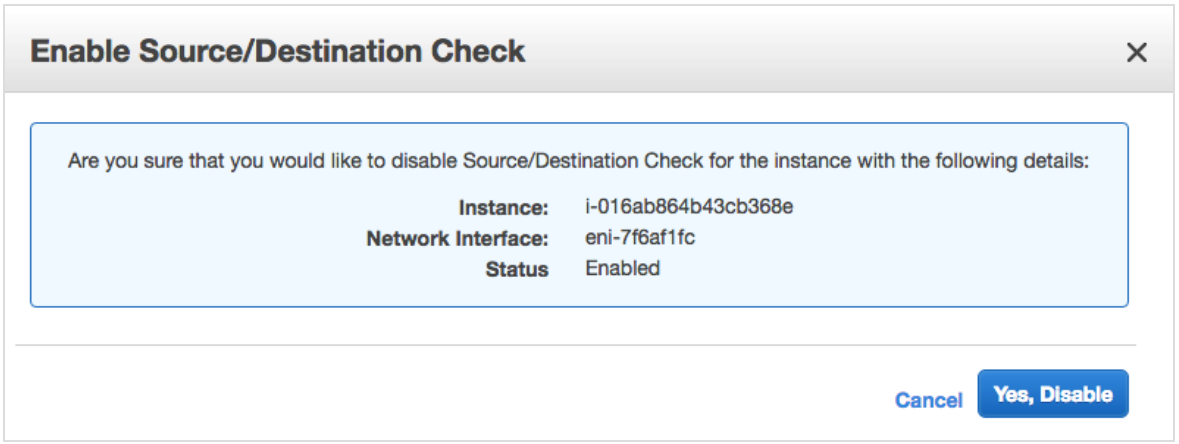
Value and Variable Used in the Config File Template	Source	Example Value and Notes
Oracle VPN headend IPSec tunnel endpoint (one per tunnel) `${ipAddress1}` `${ipAddress2}`	Oracle Console or API	Tunnel 1: 129.146.12.53 Tunnel 2: 129.146.13.53 The IP address that Oracle assigns for each tunnel when you create the IPSec connection between the DRG and CPE (see Configure Oracle Cloud Infrastructure DRG and CPE).
Oracle VPN tunnel shared secret (one per tunnel) `${psk1}` `${psk2}`	Oracle Console or API	Tunnel 1: EXAMPLEDPfAMkD7nTH3SWr6OFabdT6exXn6enSlsKbE Tunnel 2: EXAMPLEDfeSD6VNe7OfelTIKfjbxWfejPniQkNDpfjh7dwL The IPSec IKE pre-shared-key that Oracle assigns for each tunnel when you create the IPSec connection between the DRG and CPE (see Configure Oracle Cloud Infrastructure DRG and CPE).
Libreswan VTI labels (one per tunnel) `${vti1}` `${vti2}`	User	Tunnel 1: vti01 Tunnel 2: vti02 The name you assign to each tunnel's virtual tunnel interface. Each value must be unique on your Libreswan host.

Start the AWS Libreswan Configuration

1. Using the AWS Console or APIs, create a Libreswan VM by using its provisioning process. Use Oracle Linux, CentOS, or Red Hat as the main operating system.
2. After the new instance starts, connect to it with SSH and install the Libreswan package:

```
sudo yum -y install libreswan
```

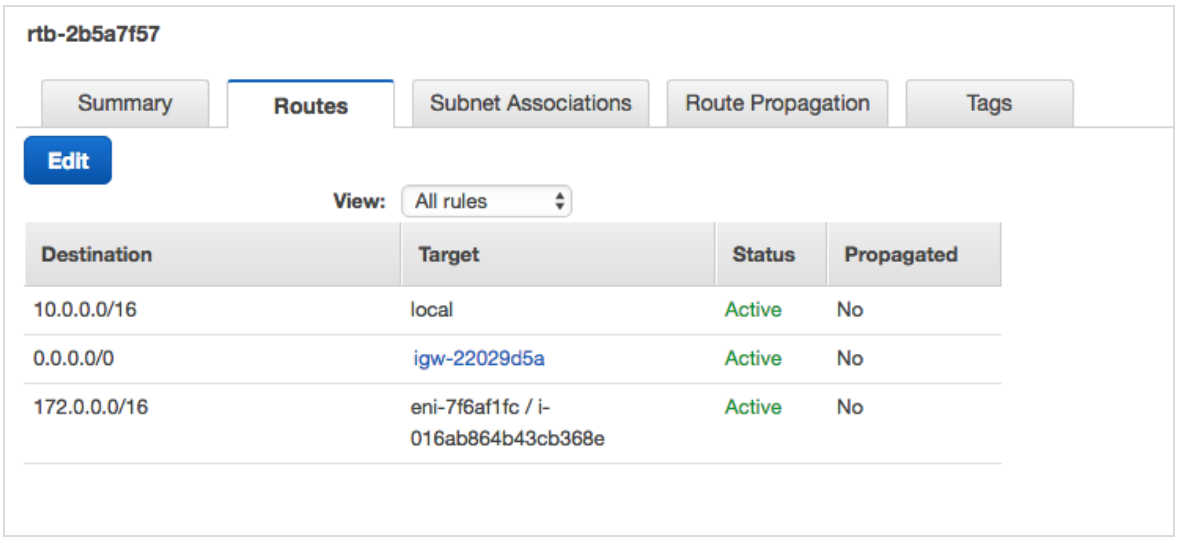
3. In the AWS Console, disable the source and destination checks on the Libreswan VM instance by right-clicking the instance, clicking **Networking**, and then clicking **Change Source/Dest. Check**. When prompted, click **Yes, Disable.**



4. On the Libreswan VM, configure IP_forward to allow AWS clients to send and receive traffic through the Libreswan VM. In the `/etc/sysctl.conf` file, set the following values and apply the updates with `sudo sysctl -p`.

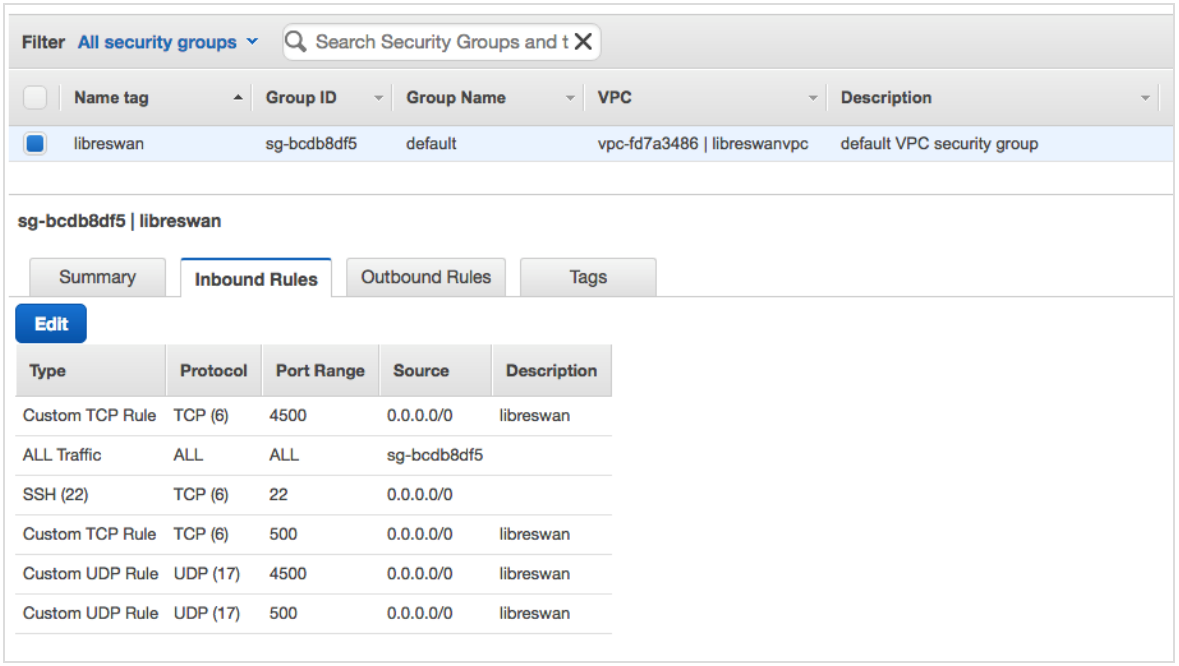
```
net.ipv4.ip_forward=1
net.ipv4.conf.all.accept_redirects = 0
net.ipv4.conf.all.send_redirects = 0
net.ipv4.conf.default.send_redirects = 0
net.ipv4.conf.eth0.send_redirects = 0
net.ipv4.conf.default.accept_redirects = 0
net.ipv4.conf.eth0.accept_redirects = 0
```

5. In the AWS Console, edit your AWS route table. Add a rule with the VCN CIDR (172.0.0.0/16) as the destination and the AWS Libreswan instance ID (i-016ab864b43cb368e in this example) as the target.

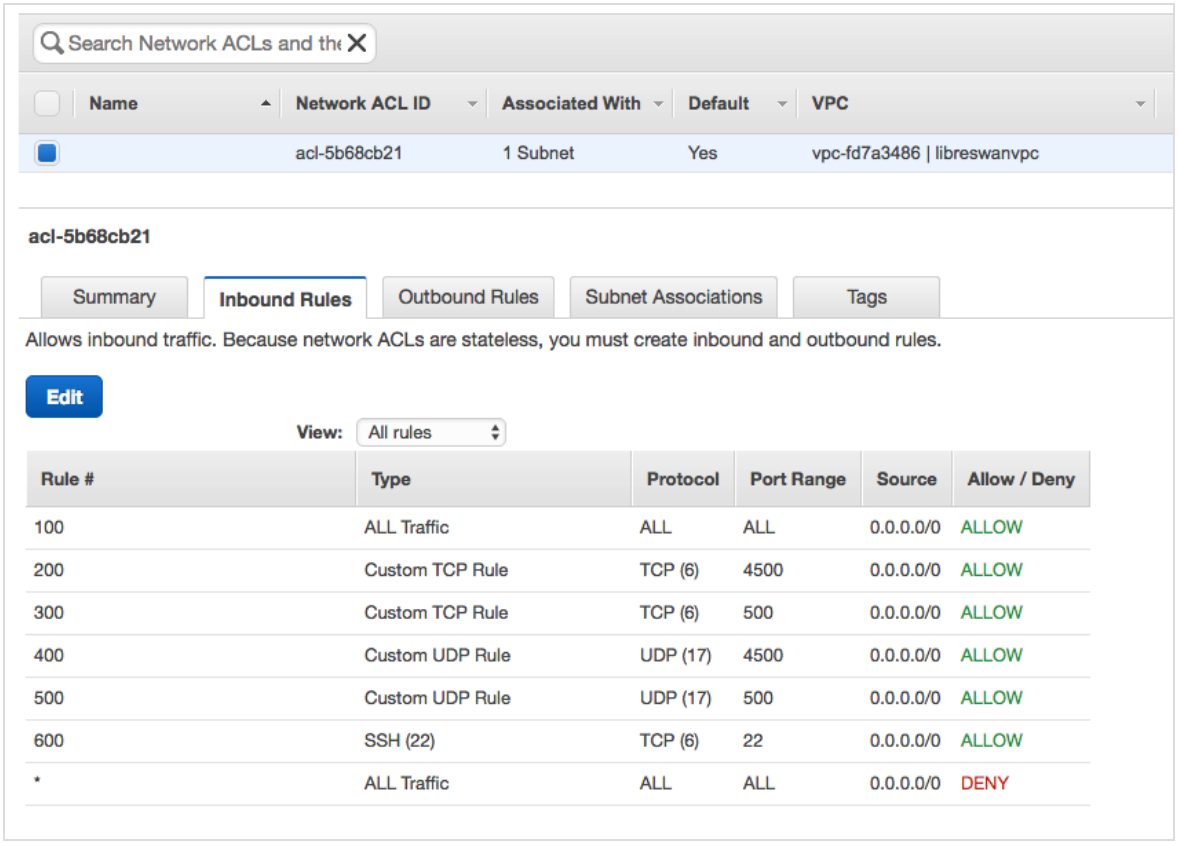


6. In the AWS Console, enable inbound TCP and UDP traffic on ports 4500 and 500 to allow Oracle Cloud Infrastructure IPSec VPN communication with the AWS Libreswan VM. This task includes editing both the AWS security groups and network ACLs. You can set the source value can be the Oracle public IP (the Oracle VPN headend IPSec tunnel endpoint) instead of 0.0.0.0/0.

For security groups:



For network ACLs:



7. Create the Libreswan IPSec configuration file:

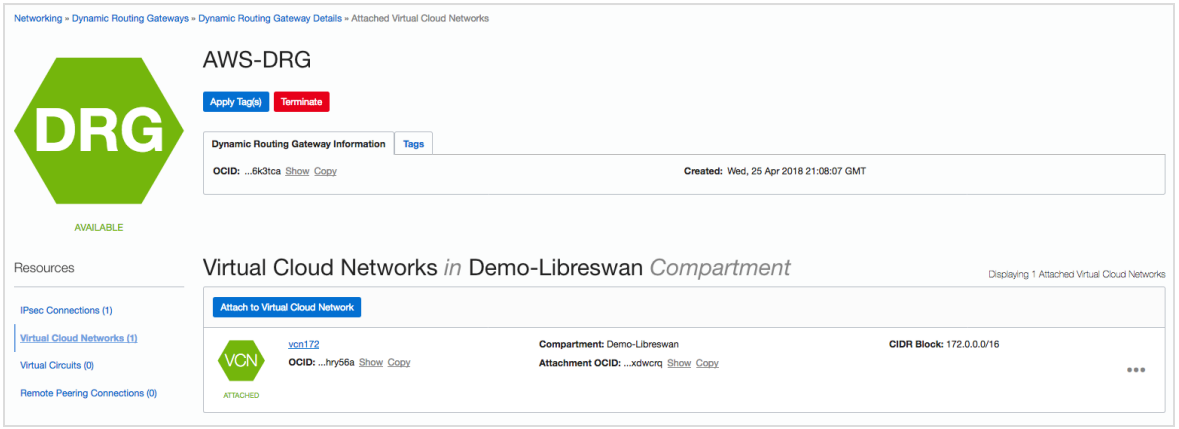
```
sudo mv /etc/ipsec.conf /etc/ipsec.conf.bck
sudo vi /etc/ipsec.conf and include the following:
config setup
include /etc/ipsec.d/*.conf
```

Configure Oracle Cloud Infrastructure DRG and CPE

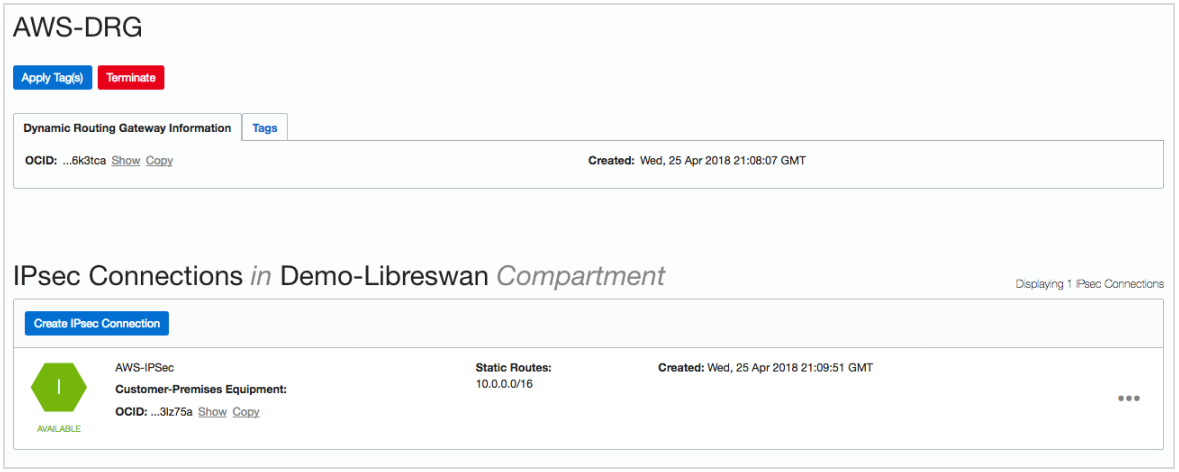
1. In the Oracle Console, create a customer-premises equipment (CPE) object that points to the Libreswan AWS instance public IP address (34.200.255.174).



2. If you don't already have a DRG attached to your VCN: in the Oracle Console, create a DRG and then attach it to the VCN (172.0.0.0/16).



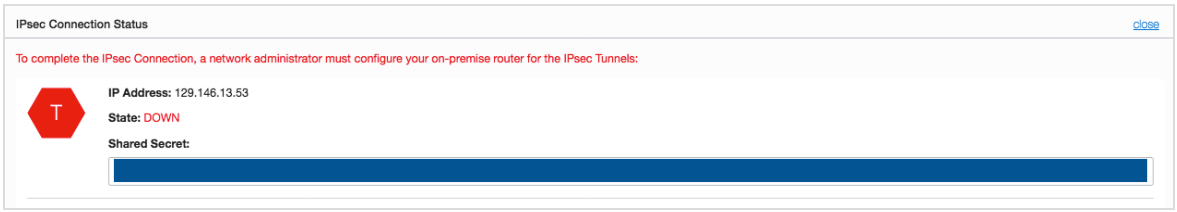
3. In the Oracle Console, create an IPSec connection and point it to the AWS VPC CIDR (10.0.0.0/16). In other words, when you create the IPSec connection, set its static route to the AWS VPC CIDR.



Oracle creates both tunnels and assigns these items to each one:

- Oracle VPN headend IPSec tunnel endpoint (129.146.13.53 in the following image, which shows only one of the tunnels)
- Oracle VPN tunnel shared secret (redacted in the following image)

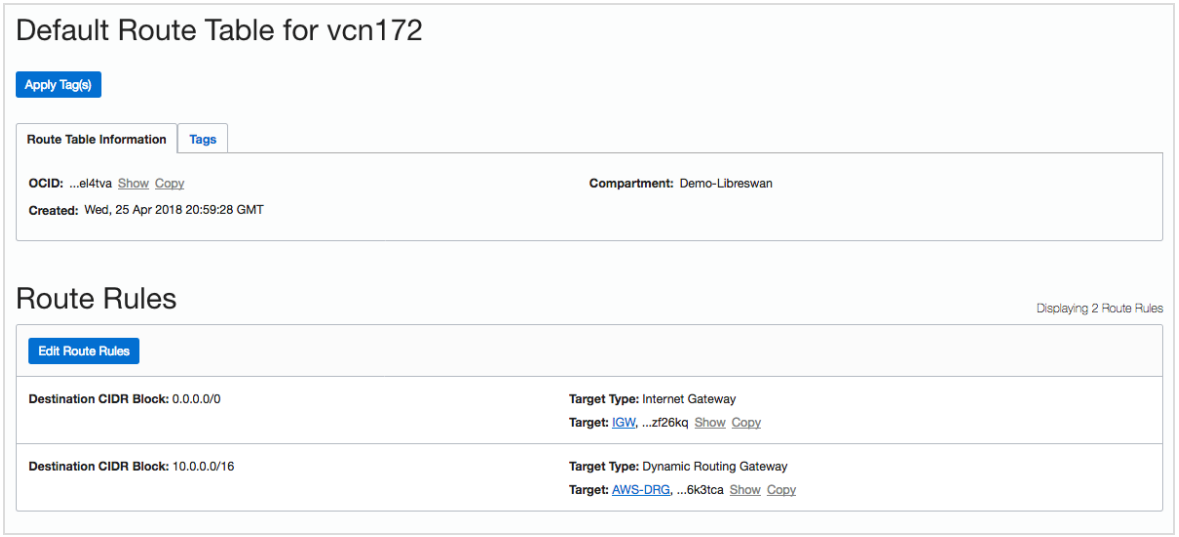
You can view these values by clicking the Actions icon (three dots) for the IPSec connection, and then clicking **Tunnel Information**. Initially each tunnel is in the DOWN state (offline) because you still have some additional configuration to do later on the AWS Libreswan VM.



4. In the Oracle Console, edit the VCN's security lists to enable ingress TCP and UDP traffic on ports 4500 and 500 like you did the AWS security groups and network ACLs. You can use the AWS Libreswan VM public IP address instead of 0.0.0.0/0 if it's a persistent public IP. Also open all protocols and ports for ingress traffic from the AWS VPC CIDR (10.0.0.0/16). Remember: Security lists are associated with a subnet, so edit the security list associated with each subnet that needs to communicate with the AWS VPC.

Source: 10.0.0.0/16	**IP Protocol:** All Protocols			**Allows:** all traffic for all ports
Source: 0.0.0.0/0	**IP Protocol:** TCP	**Source Port Range:** All	**Destination Port Range:** 4500	**Allows:** TCP traffic for ports: 4500
Source: 0.0.0.0/0	**IP Protocol:** TCP	**Source Port Range:** All	**Destination Port Range:** 500	**Allows:** TCP traffic for ports: 500
Source: 0.0.0.0/0	**IP Protocol:** UDP	**Source Port Range:** All	**Destination Port Range:** 4500	**Allows:** UDP traffic for ports: 4500
Source: 0.0.0.0/0	**IP Protocol:** UDP	**Source Port Range:** All	**Destination Port Range:** 500	**Allows:** UDP traffic for ports: 500

5. In the Oracle Console, edit the VCN's route tables to add a rule that has the AWS VPC CIDR (10.0.0.0/16) as the destination CIDR block and the DRG you created earlier as the target. Remember: Route tables are associated with a subnet, so edit the route table associated with each subnet that needs to communicate with the AWS VPC. The following screenshot shows the default route table for the VCN.



Set Up Your Configuration File: /etc/ipsec.d/oci-ipsec.conf

Use the following template for your `/etc/ipsec.d/oci-ipsec.conf` file. The file defines the two tunnels that Oracle creates when you set up the IPSec connection. Replace the variables such as `${ipAddress1}` with your specific values (see the table in Information to Gather).

Important

If your CPE is behind a 1-1 NAT device, uncomment the `rightid` parameter and set it equal to the `${cpePublicIpAddress}`.

```
conn oracle-tunnel-1
     left=${cpeLocalIP}
     # leftid=${cpePublicIpAddress} # See preceding note about 1-1 NAT device
     right=${ipAddress1}
     authby=secret
     leftsubnet=0.0.0.0/0
     rightsubnet=0.0.0.0/0
     auto=start
     mark=5/0xffffffff # Needs to be unique across all tunnels
     vti-interface=${vti1}
     vti-routing=no
     encapsulation=no
     ikelifetime=28800s
     salifetime=3600s
conn oracle-tunnel-2
     left=${cpeLocalIP}
     # leftid=${cpePublicIpAddress} # See preceding note about 1-1 NAT device
     right=${ipAddress2}
     authby=secret
     leftsubnet=0.0.0.0/0
     rightsubnet=0.0.0.0/0
     auto=start
     mark=6/0xffffffff # Needs to be unique across all tunnels
     vti-interface=${vti2}
     vti-routing=no
     encapsulation=no
     ikelifetime=28800s
     salifetime=3600s
```

Set Up Your Secrets File: /etc/ipsec.d/oci-ipsec.secrets

Use the following template for your `/etc/ipsec.d/oci-ipsec.secrets` file. It contains two lines per IPSec connection (one line per tunnel).

```
${cpePublicIpAddress} ${ipAddress1}: PSK "${psk1}"
${cpePublicIpAddress} ${ipAddress2}: PSK "${psk2}"
```

Reload the Libreswan Configuration

After setting up your configuration and secrets files, you must restart the Libreswan service.



Important

Restarting the Libreswan service may impact existing tunnels.

The following command rereads the config file and restarts the Libreswan service. If you're logged in with an unprivileged user account, you might need to use `sudo` before the command.

```
service ipsec restart
```

Check the Libreswan Status

Check the current state of your Libreswan tunnels by using the following command.

```
ipsec status
```

The tunnel is established if you see a line that includes the following:

```
STATE_MAIN_I4: ISAKMP SA established
```

In the future, if you need to open a support ticket with Oracle about your Libreswan tunnel, include the output of the preceding `ipsec status` command.

Check the Tunnel Interface Status

Check if the virtual tunnel interfaces are up or down by using the `ifconfig` command or the `ip link show` command. You can also use applications such as tcpdump with the interfaces.

Here's an example of the `ifconfig` output:

```
ifconfig
<output trimmed>

vti01: flags=209<UP,POINTOPOINT,RUNNING,NOARP> mtu 8980
     inet6 fe80::5efe:a00:2 prefixlen 64 scopeid 0x20<link>
     tunnel txqueuelen 1000 (IPIP Tunnel)
     RX packets 0 bytes 0 (0.0 B)
     RX errors 0 dropped 0 overruns 0 frame 0
     TX packets 0 bytes 0 (0.0 B)
     TX errors 10 dropped 0 overruns 0 carrier 10 collisions 0

vti02: flags=209<UP,POINTOPOINT,RUNNING,NOARP> mtu 8980
     inet6 fe80::5efe:a00:2 prefixlen 64 scopeid 0x20<link>
     tunnel txqueuelen 1000 (IPIP Tunnel)
     RX packets 0 bytes 0 (0.0 B)
     RX errors 0 dropped 0 overruns 0 frame 0
     TX packets 0 bytes 0 (0.0 B)
     TX errors 40 dropped 0 overruns 0 carrier 40 collisions 0
```

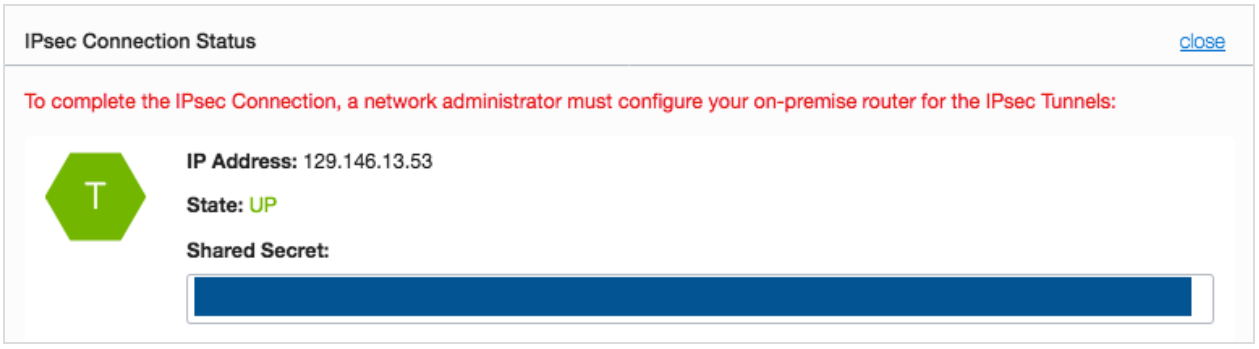
Here's an example of the `ip link show` output:

```
ip link show
<output trimmed>

9: vti01@NONE: <POINTOPOINT,NOARP,UP,LOWER_UP> mtu 8980 qdisc noqueue
state UNKNOWN mode DEFAULT group default qlen 1000
   link/ipip 10.0.0.2 peer 129.146.12.53

10: vti02@NONE: <POINTOPOINT,NOARP,UP,LOWER_UP> mtu 8980 qdisc noqueue
state UNKNOWN mode DEFAULT group default qlen 1000
   link/ipip 10.0.0.2 peer 129.146.13.53
```

Also, in the Oracle Cloud Infrastructure Console, each IPSec tunnel should now be in the UP state.



Configure IP Routing

Use the following `ip` command to create static routes that send traffic to your VCN through the IPSec tunnels. If you're logged in with an unprivileged user account, you might need to use `sudo` before the command.

```
ip route add ${VcnCidrBlock} nexthop dev ${vti1} nexthop dev ${vti2}
ip route show
```

Test the Connection

Now that the tunnels are up, you can test whether an Oracle Cloud Infrastructure VM can communicate with an AWS VM over the IPSec connection. One easy way is with the ping command. The following example shows a successful ping from each side of the connection. The VMs from both cloud providers can communicate with each other.

Left Side: Oracle Cloud Infrastructure VM

Public IP: 129.146.74.114
VCN Local IP: 172.0.0.10

```
[opc@ociclient172 ~]$ ifconfig ens3
ens3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 9000
        inet 172.0.0.10  netmask 255.255.255.0  broadcast 172.0.0.255
        inet6 fe80::200:17ff:fe01:b7bd  prefixlen 64  scopeid 0x20<link>
        ether 00:00:17:01:b7:bd  txqueuelen 1000  (Ethernet)
        RX packets 60693  bytes 264662540 (252.4 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 53902  bytes 130659925 (124.6 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[opc@ociclient172 ~]$ ping 10.0.0.11 -c 3
PING 10.0.0.11 (10.0.0.11) 56(84) bytes of data.
64 bytes from 10.0.0.11: icmp_seq=1 ttl=63 time=53.4 ms
64 bytes from 10.0.0.11: icmp_seq=2 ttl=63 time=53.4 ms
64 bytes from 10.0.0.11: icmp_seq=3 ttl=63 time=53.3 ms

--- 10.0.0.11 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 53.312/53.417/53.496/0.077 ms
[opc@ociclient172 ~]$
```

Right Side: AWS VM

Public IP: 34.201.24.5
VPC Local IP: 10.0.0.11

```
[centos@ip-10-0-0-11 ~]$ ifconfig eth0
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 9001
        inet 10.0.0.11  netmask 255.255.255.0  broadcast 10.0.0.255
        inet6 fe80::fc:61ff:feba:d7e0  prefixlen 64  scopeid 0x20<link>
        ether 02:fc:61:ba:d7:e0  txqueuelen 1000  (Ethernet)
        RX packets 10976  bytes 824442 (805.1 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 10981  bytes 1297987 (1.2 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[centos@ip-10-0-0-11 ~]$ ping -c 3 172.0.0.10
PING 172.0.0.10 (172.0.0.10) 56(84) bytes of data.
64 bytes from 172.0.0.10: icmp_seq=1 ttl=62 time=53.5 ms
64 bytes from 172.0.0.10: icmp_seq=2 ttl=62 time=53.3 ms
64 bytes from 172.0.0.10: icmp_seq=3 ttl=62 time=53.4 ms

--- 172.0.0.10 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2001ms
rtt min/avg/max/mdev = 53.334/53.452/53.569/0.211 ms
[centos@ip-10-0-0-11 ~]$
```

Network Performance

The content in the sections below apply to **Category 7** and **Section 3.c** of the Oracle PaaS and IaaS Public Cloud Services Pillar documentation, which you can download in PDF format from the Oracle Cloud Infrastructure Service Level Agreement page.

Oracle Cloud Infrastructure provides a service-level agreement (SLA) for network throughput between instances in the same availability domain in a virtual cloud network (VCN). You might think of this as a measurement of LAN performance.

Important

This SLA applies only to bare metal instances.

To meet the SLA, the network throughput for instances within the same availability domain and VCN must be at least 90% of the stated maximum for at least 99.9% of the billing month. Network throughput is measured in megabits per second (Mbps) or gigabits per second (Gbps).

For the stated maximum bandwidth by instance shape, see the "Network Bandwidth" column in the "Shape" tables.

Testing Methodology

Summary: Launch two bare metal instances in the same availability domain and VCN. Install and run the `iperf3` utility, with one instance as server and the other as client. Look at the `iperf3` bandwidth results to determine your VCN's network throughput.

Instructions:

1. Launch two bare metal instances in the same availability domain in a single VCN. Designate one as the server and the other as the client. For launch instructions, see Creating an Instance.
2. Install `iperf3` on both instances. Example Linux command:

   ```
   sudo yum install -y iperf3
   ```

3. Enable communication to the server instance on TCP port 5201 (for `iperf3`):
   a. For the subnet that the server instance is in, add a rule to the subnet's security list to allow stateless ingress traffic on TCP port 5201 from any source IP address (0.0.0.0/0) and any source port. For instructions, see To update rules in an existing security list.
   b. On the instance itself, open the firewall to allow `iperf3` traffic. Example Linux commands:

      ```
      sudo firewall-cmd --zone=public --permanent --add-port 5201/tcp

      sudo firewall-cmd --reload
      ```

4. Start the `iperf3` test:
    a. On the server instance, run `iperf3` in server mode. Example Linux command:

    ```
    iperf3 -s
    ```

    b. On the client instance, run `iperf3` in client mode and specify the private IP address of the server instance. Example Linux command:

    ```
    iperf3 -c <server_instance_private_ip_address>
    ```

5. Look at the `iperf3` results on the client instance. The network throughput between the two instances is shown under "Bandwidth" in the last five lines of the client's `iperf3` test output. For example:

```
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bandwidth       Retr
[  4] 0.00-10.00  sec  XX.YY GBytes  NN.NN Gbits/sec  752             sender
[  4] 0.00-10.00  sec  XX.YY GBytes  NN.NN Gbits/sec                  receiver

iperf Done.
```

Frequently Asked Questions

Q: My VCN isn't meeting the bandwidth SLA. What should I do?

A: Make sure that the CPU on the instance isn't loaded heavily with other services or applications. Confirm this with a utility such as `top` to look at the average CPU utilization. It should be less than one.

Networking Metrics

You can monitor the health, capacity, and performance of your Networking service VNICs by using metrics, alarms, and notifications.

This topic describes the metrics emitted by the metric namespace `oci_vcn` (the Networking service).

Resources: virtual network interface cards (VNICs).

Overview of Metrics for an Instance and Its Network Devices

If you're not already familiar with the different types of metrics available for an instance and its storage and network devices, see Compute Metrics.

Overview of Metrics: oci_vcn

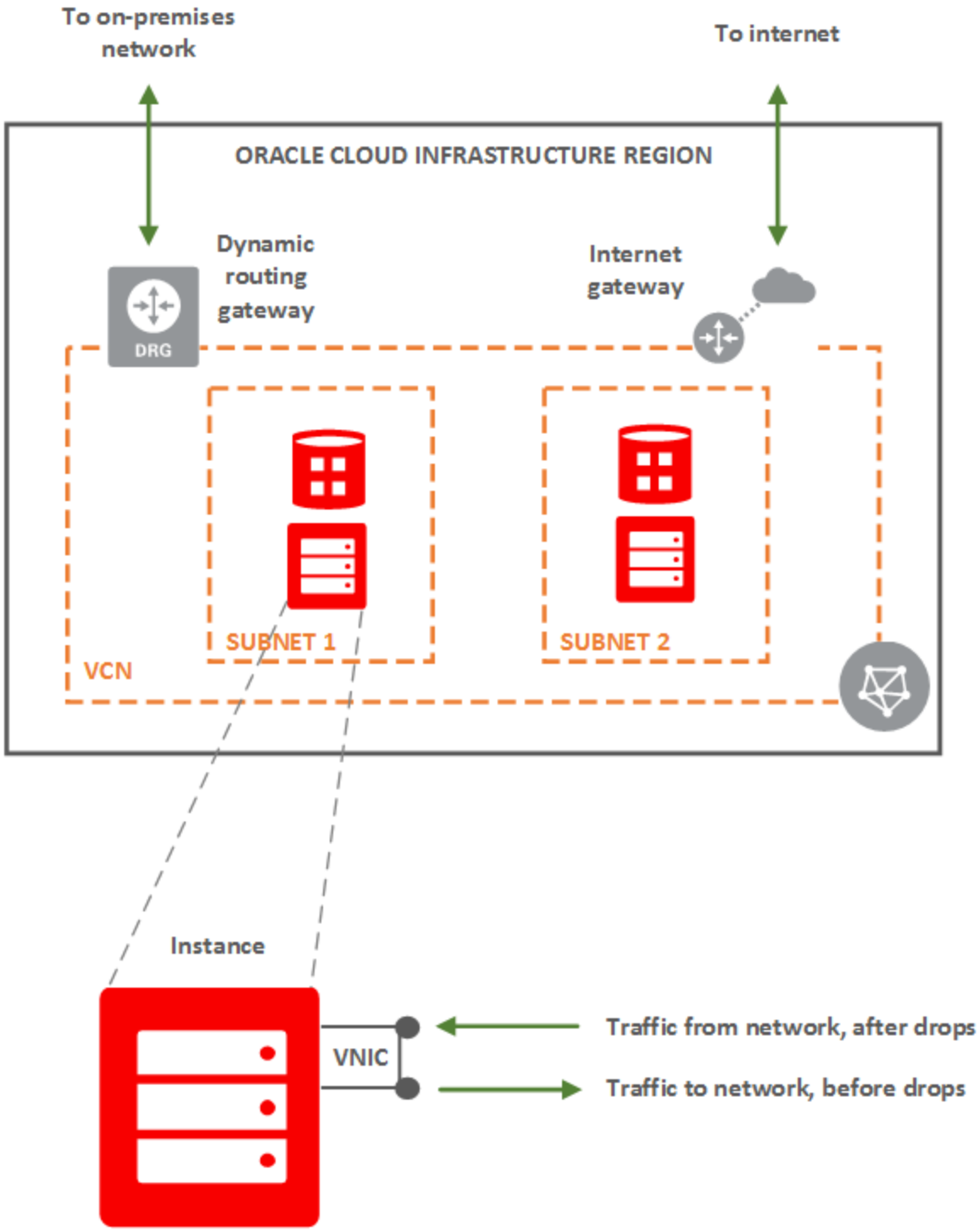
Each Compute instance has one or more Networking service VNICs. A VNIC connects the instance to a subnet in a virtual cloud network (VCN). A given VNIC controls how the instance communicates with endpoints inside the VCN (other instances) and endpoints outside the VCN (hosts on the internet, in your on-premises network, in another VCN, and so on).

With the Networking service metrics (in metric namespace `oci_vcn`), you can get this information for a VNIC:

- **Traffic to and from the network:** Per-VNIC traffic levels (packets and bytes), which can help you identify meaningful increases or decreases in traffic coming in and out of your instances
- **Packets dropped due to security list violations:** Per-VNIC *drops* (dropped packets), which can help you identify changes in traffic caused by security list changes

The following diagram illustrates the general concept. A given instance resides in a subnet within a VCN that has one or more gateways to communicate with other networks. The instance is enlarged to show its VNIC, which the instance uses to communicate with the network. In this context, the term *network* means both the other instances in the VCN and hosts outside the VCN available through the gateways.

The VNIC receives traffic from the network and sends traffic to the network. The Networking service drops packets according to security list rules you set up for the instance's subnet. Traffic coming to the VNIC from the network is measured *after* the Networking service drops the packets that violate the subnet's security list rules. Traffic leaving the VNIC is measured *before* the Networking service drops the packets that violate the subnet's security list rules.


To on-premises network
To internet
ORACLE CLOUD INFRASTRUCTURE REGION
Dynamic routing gateway
DRG
Internet gateway
SUBNET 1
SUBNET 2
VCN
Instance
VNIC
Traffic from network, after drops
Traffic to network, before drops

The Compute service separately reports network-related metrics *as measured on the instance itself and aggregated across all the attached VNICs*. Those metrics are available in the `oci_computeagent` metric namespace. For more information, see Compute Metrics.

Required IAM Policy

When writing an IAM policy for viewing VNIC metrics, it's important to remember that:

- The VNIC and the VNIC's metrics (emitted by the `oci_vcn` metric namespace) reside in the *subnet's compartment*, and not the instance's compartment.
- The *VNIC attachment* (which is an object different from the VNIC itself) resides in the *instance's compartment*.

If the instance and subnet are in the same compartment, these details aren't so important when you write the IAM policy.

Minimum required policy for getting VNIC metrics

The following policy contains the one statement required to get VNIC metrics, which are part of the `oci_vcn` metric namespace.

If you're using the Console, this policy lets you go to the **Monitoring** tab in the Console and view the metrics for one or more VNICs in the specified compartment. The policy uses an example group called VnicMetricReaders. The condition at the end (`where target.metrics.namespace='oci_vcn'`) allows the group to view only the metrics in the `oci_vcn` metric namespace.

```
Allow group VnicMetricReaders to read metrics in compartment <subnet_compartment> where
target.metrics.namespace='oci_vcn'
```

Policy for viewing a VNIC's details and metrics in the Console

The following policy lets you view an instance in the Console, click through to a specific VNIC, and then view that VNIC's details and metrics.

```
Allow group VnicMetricReaders to read metrics in compartment <subnet_compartment> where
target.metrics.namespace='oci_vcn'

Allow group VnicMetricReaders to read instance-family in compartment <instance_compartment>

Allow group VnicMetricReaders to inspect virtual-network-family in compartment <subnet_compartment>
```

The second and third statements let you view the instance's details and the VNIC's details, respectively.

Available Metrics: oci_vcn

The metrics listed in the following table are automatically available for any VNIC on any instance you create. You do not need to enable monitoring on the instance to get these metrics for the VNIC or VNICs on the instance.

You also can use the Monitoring service to create custom queries.

Each metric includes the following dimension:

RESOURCEID

The OCID of the VNIC.

Metric	Metric Display Name	Unit	Description	Dimensions
VnicEgressDropsSecurityList	**Egress Packets Dropped by Security List**	packets	Packets sent by the VNIC, destined for the network, dropped due to security rule violations.	resourceId
VnicIngressDropsSecurityList	**Ingress Packets Dropped by Security List**	packets	Packets received from the network, destined for the VNIC, dropped due to security rule violations.	
VnicFromNetworkBytes*	**Bytes from Network**	bytes	Bytes received at the VNIC from the network, after drops.	
VnicFromNetworkPackets*	**Packets from Network**	packets	Packets received at the VNIC from the network, after drops.	

Metric	Metric Display Name	Unit	Description	Dimensions
`VnicToNetworkBytes`*	**Bytes to Network**	bytes	Bytes sent from the VNIC to the network, before drops.	
`VnicToNetworkPackets`*	**Packets to Network**	packets	Packets sent from the VNIC to the network, before drops.	

* The Compute service separately reports network-related metrics *as measured on the instance itself and aggregated across all the attached VNICs*. Those metrics are available in the `oci_computeagent` metric namespace. For more information, see Compute Metrics.

Using the Console

To view default metric charts for a single VNIC

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
2. Click the instance to view its details.
3. Click **Attached VNICs**.
4. Click the VNIC to view its details.
5. Under **Resources**, click **Metrics**.

For more information about monitoring metrics and using alarms, see Monitoring Overview. For information about notifications for alarms, see Notifications Overview.

To view default metric charts for multiple VNICs

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Service Metrics**.
2. For **Compartment**, select the compartment that contains the VNICs you're interested in. Remember that a given VNIC resides in its *subnet's* compartment.
3. For **Metric Namespace**, select **oci_vcn**.
   The **Service Metrics** page dynamically updates the page to show charts for each metric that is emitted by the selected metric namespace.



Tip

If there are multiple VNICs in the compartment, the charts default to show a separate line for each VNIC. You can instead show a single line aggregated across all the VNICs by selecting the check box for **Aggregate Metric Streams** on the right side of the page.

For more information about monitoring metrics and using alarms, see Monitoring Overview. For information about notifications for alarms, see Notifications Overview.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following APIs for monitoring:

- Monitoring API for metrics and alarms
- Notifications API for notifications (used with alarms)

Troubleshooting

These topics cover some common issues you might run into and how to address them:

- Hanging Connection
- Subnet or VCN Deletion
- VPN Connect Troubleshooting
- FastConnect Troubleshooting

Hanging Connection

This topic covers one of the most common issues seen with communications between your cloud network and on-premises network: a hanging connection, even though you can ping hosts across the connection.

Summary of Problem and Solutions

Symptom: Your virtual cloud network (VCN) is connected to your existing on-premises network via an IPSec VPN, or Oracle Cloud Infrastructure FastConnect. Hosts on one side of the connection can ping hosts on the other side, but the connection hangs. For example:

- You can SSH to a host across the connection, but after you log in to the host, the connection hangs.
- You can start a Virtual Networking Computing (VNC) connection, but the session hangs.
- You can start an SFTP download, but the download hangs.

General problem: *Path Maximum Transmission Unit Discovery (PMTUD)* is probably not working on one or both sides of the connection. It must be working on both sides of the connection so that both sides can know if they're trying to send packets that are too large for the connection and adjust accordingly. For a brief overview of Maximum Transmission Unit (MTU) and PMTUD, see Overview of MTU and Overview of PMTUD.

Solutions for fixing PMTUD:

1. **Make sure your hosts are configured to use PMTUD:** If the hosts in your on-premises network don't use PMTUD (that is, if they don't set the Don't Fragment flag in the packets), they have no way to discover if they're sending packets that are too large for the connection. Your instances on the Oracle side of the connection use PMTUD by default. Do not change that configuration on the instances.
2. **Make sure both the VCN security lists and the instance firewalls allow ICMP type 3 code 4 messages:** When PMTUD is in use, the sending hosts receive a special ICMP message if they send packets that are too large for the connection. Upon receipt of the message, the host can dynamically update the size of the packets to fit the connection. However, your instances can't receive these important ICMP messages if the security lists for the subnet in the VCN and/or the instance firewalls aren't configured to accept them.

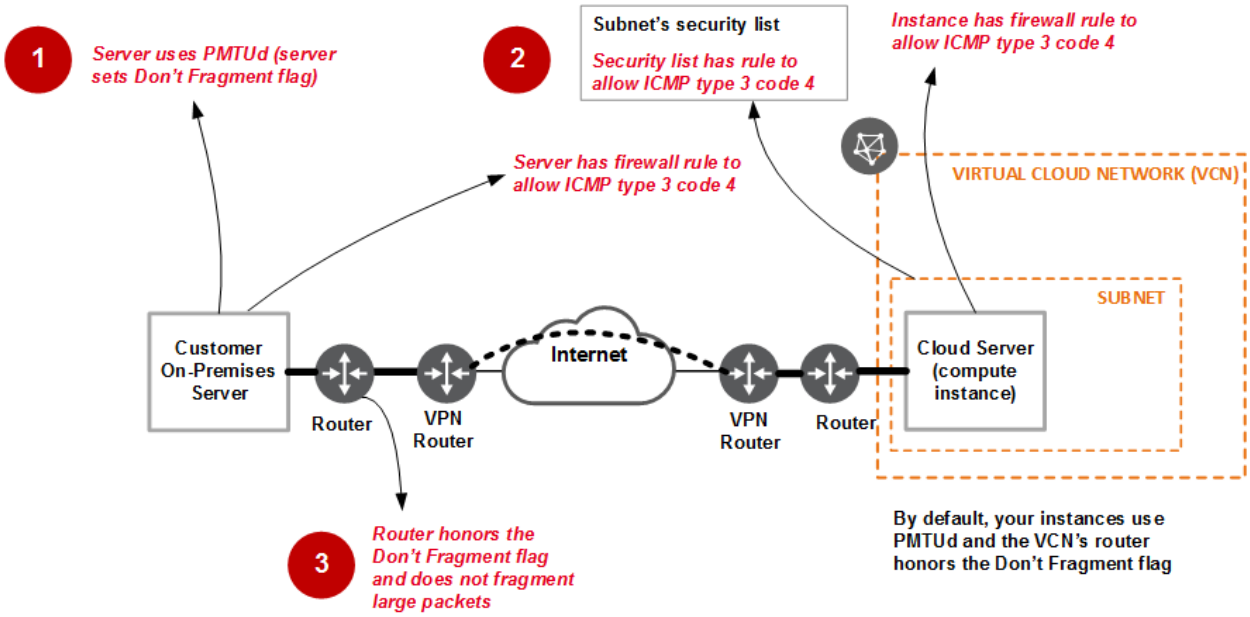
   Tip

   If you're using stateful security list rules (for TCP, UDP, or ICMP traffic), you don't need to ensure that your security list has a rule to allow ICMP type 3 code 4 messages. With stateful rules, the Networking service tracks the connections and automatically allows corresponding ICMP type 3 code 4 messages without needing an explicit rule to allow them. Only if you're using stateless rules must you have an explicit ingress security list rule for ICMP type 3 code 4 messages. You must also confirm the instance firewalls are set up correctly.

   To check to see if a host is receiving the messages, see Finding Where PMTUD Is Broken.
3. **Make sure your router honors the *Don't Fragment* flag:** If the router doesn't honor the flag and thus ignores the use of PMTUD, it sends fragmented packets to the

instances in the VCN, which is bad (see Why Avoid Fragmentation?). The VCN's security lists are most likely configured in such a way that they recognize only the initial fragment, and the remaining ones are dropped, causing the connection to hang. Instead, your router should use PMTUD and honor the Don't Fragment flag to determine the correct size of unfragmented packets to send through the connection.

The parts of the solution are numbered and called out in red italics in the following diagram. It shows an example scenario with your on-premises network connected to your VCN over an IPSec VPN.



Keep reading for a brief overview of MTU and PMTUD, and how to check if PMTUD is working on both sides of the network connection.

Why Avoid Fragmentation?

You may be wondering why you want to avoid fragmentation. First, it adversely affects the performance of your application. Fragmentation requires reassembly of the fragments and

retransmission if fragments are lost. Reassembly and retransmission require time and CPU resources.

Second, only the first fragment contains the source and destination port information. This means the other packets will probably be dropped by firewalls or your VCN's security lists, which are typically configured to evaluate the port information. For fragmentation to work with your firewalls and security lists, you would have to configure them to be more permissive than usual, which is not desirable.

Overview of MTU

The communications between any two hosts across an Internet Protocol (IP) network use packets. Each packet has a source and destination IP address and a payload of data. Every network segment between the two hosts has a *Maximum Transmission Unit (MTU)* that represents the number of bytes that a single packet can carry.

Across the internet, the MTU is 1500 bytes. This is also true for most home networks and many corporate networks (and their Wi-Fi networks). Some data centers, including those for Oracle Cloud Infrastructure, have a larger MTU. The Compute instances use an MTU of 9000 by default. On a Linux host, you can use the `ifconfig` command to display the MTU of the host's network connection. For example, here's the `ifconfig` output from an Ubuntu instance (the MTU is highlighted in red italics):

```
ifconfig
ens3 Link encap:Ethernet HWaddr 00:00:17:01:17:83
inet addr:10.0.6.9 Bcast:10.0.6.31 Mask:255.255.255.224
inet6 addr: fe80::200:17ff:fe01:1783/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:9000 Metric:1
```

For comparison, here's the output from a machine connected to a corporate network:

```
ifconfig
en0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST>
mtu 1500
```

Notice that its MTU is the more typical 1500 bytes.

If the host is connected through a corporate VPN, the MTU is even smaller, because the VPN tunnel must encapsulate the traffic inside an IPSec packet and send it across the local network. For example:

```
ifconfig
utun0: flags=81d1<UP,POINTOPOINT,RUNNING,NOARP,PROMISC,MULTICAST>
mtu 1300
```

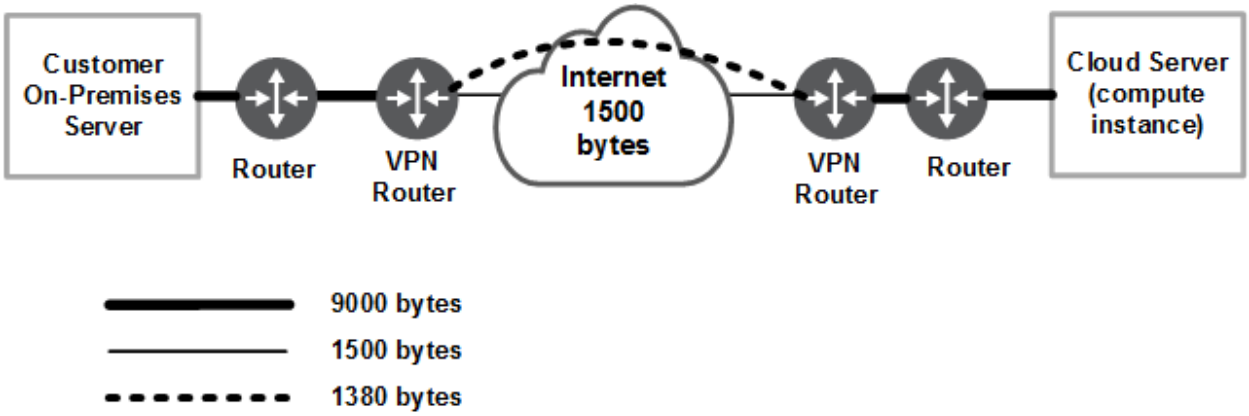
How do the two hosts figure out how large of a packet they can send to each other? For many types of network traffic, such as HTTP, SSH, and FTP, the hosts use the Transmission Control Protocol (TCP) to establish new connections. During the initial three-way handshake between two hosts, they each send the *Maximum Segment Size (MSS)* for how large their payload can be. This is smaller than the MTU. (TCP runs inside the Internet Protocol (IP), which is why it's referred to as TCP/IP. Segments are to TCP what packets are to IP.)

Using the tcpdump application, you can see the MSS value shared during the handshake. Here's an example from tcpdump (with the MSS highlighted in red italics):

```
12:11:58.846890 IP 129.146.27.25.22 > 10.197.176.19.58824: Flags [S.], seq
2799552952, ack 2580095593, win 26844, options [mss 1260,sackOK,TS val
44858491 ecr 1321638674,nop,wscale 7], length 0
```

The preceding packet is from an SSH connection to an instance from a laptop connected to a corporate VPN. The local network the laptop uses for its internet connection has an MTU of 1500 bytes. The VPN tunnel enforces an MTU of 1300 bytes. Then when the SSH connection is attempted, TCP (running inside the IP connection) tells the Oracle Cloud Infrastructure instance that it supports TCP segments that are less than or equal to 1260 bytes. With a corporate VPN connection, the laptop connected to the VPN typically has the smallest MTU and MSS compared to anything it's communicating with across the internet.

A more complex case is when the two hosts have a larger MTU than some network link between them *that is not directly connected to either of them*. The following diagram illustrates an example.



In this example, there are two servers, each directly connected to its own routed network that supports a 9000-byte MTU. The servers are in different data centers. Each data center is connected to the internet, which supports a 1500-byte MTU. There is an IPSec VPN tunnel between the two data centers. That tunnel crosses the internet, so the inside of the tunnel has a smaller MTU than the internet. In this diagram, the MTU is 1380 bytes.

If the two servers try to communicate (with SSH, for example), during the three-way handshake, they agree on an MSS around 8960. The initial SSH connection might succeed, because the maximum packet sizes during the initial SSH connection setup are usually less than 1380 bytes. When one side tries to send a packet larger than the smallest link between the two endpoints, Path MTU Discovery (PMTUD) becomes critical.

Overview of PMTUD

Path MTU Discovery is defined in RFC 1191. It works by requiring the two communicating hosts to set a *Don't Fragment* flag in the packets they each send. If a packet from one of these hosts reaches a router where the egress (or outbound) interface has an MTU smaller than the packet length, the router drops that packet. The router also returns an ICMP type 3 code 4

message to the host. This message specifically says "Destination Unreachable, Fragmentation Needed and Don't Fragment Was Set" (defined in RFC 792). Effectively the router tells the host: "You told me not to fragment packets that are too large, and this one's too large. I'm not sending it." The router also tells the host the maximum size packets allowed through that egress interface. The sending host then adjusts the size of its outbound packets so they're smaller than the value the router provided in the message.

Here's an example that shows the results when an instance tries to ping a host (8.8.8.8) over the internet with an 8000-byte packet and the Don't Fragment flag set (that is, with PMTUD in use). The returned ICMP message is highlighted in red italics:

```
ping 8.8.8.8 -M do -s 8000
PING 8.8.8.8 (8.8.8.8) 8000(8028) bytes of data.
From 4.16.139.250 icmp_seq=1 Frag needed and DF set (mtu = 1500)
```

The response is exactly what's expected. The destination host is across the internet, which has an MTU of 1500 bytes. Even though the sending host's local network connection has an MTU of 9000 bytes, the host can't reach the destination host with the 8000-byte packet and gets an ICMP message accordingly. PMTUD is working correctly.

For comparison, here's the same ping, but the destination host is across an IPSec VPN tunnel:

```
ping 192.168.6.130 -M do -s 8000
PING 192.168.6.130 (192.168.6.130) 8000(8028) bytes of data.
From 129.146.13.49 icmp_seq=1 Frag needed and DF set
```

Here the VPN router sees that to send this packet to its destination, the outbound interface is a VPN tunnel. That tunnel goes across the internet, so the tunnel must fit inside the internet's 1500-byte MTU link. The result is that the inside of the tunnel only allows packets up to 1360 bytes (which the router then lowered to 1358, which can make things more confusing).

Finding Where PMTUD Is Broken

If PMTUD isn't working somewhere along the connection, you need to figure out why and where. Typically it's because the ICMP type 3 code 4 packet (from the router with the constrained link that can't fit the packet) never gets back to the sending host. This can happen if there's something blocking that kind of traffic between the host and the router. And it can happen on either side of the VPN tunnel (or other constrained MTU link).

TRY PINGING FROM EACH SIDE OF THE CONNECTION

To troubleshoot the broken PMTUD, you must determine if PMTUD is working on each side of the connection. In this scenario, let's assume the connection is an IPSec VPN.

How to ping: Like in Overview of PMTUD, ping a host on the other side of the connection with a packet that you know is too large to fit through the VPN tunnel (for example, 1500 bytes or larger). Depending on which operating system the sending host uses, you might need to format the ping command slightly different to ensure the Don't Fragment flag is set. For both Ubuntu and Oracle Linux, you use the `-M` flag with the ping command.

Here's information about the `-M` flag:

```
-M pmtudisc_opt
Select Path MTU Discovery strategy. pmtudisc_option may be either do
(prohibit fragmentation, even local one), want (do PMTU discovery, fragment
locally when packet size is large), or dont (do not set DF flag).
```

Here's an example ping (with the -M flag and the resulting ICMP message highlighted in red italics)

```
ping -M do -s 1500 192.168.6.130
PING 192.168.6.130 (192.168.6.130) 1500(1528) bytes of data.
From 129.146.13.49 icmp_seq=1 Frag needed and DF set (mtu = 1358)
```

Good: PMTUD is working

If the result includes the line "From x.x.x.x icmp_seq=1 Frag needed and DF set (mtu = xxxx)", then PMTUD is working on that side of the tunnel. Note that the source address of the ICMP message is the public IP address of the tunnel the traffic is trying to go out (for example 129.146.13.49 in the preceding Ubuntu example).

Make sure to also ping from the other side of the connection to confirm PMTUD is working from that side. Both sides of the connection must recognize that there is a tunnel between them that can't fit the large packets.

Bad: If you're testing your side of the connection and the ping succeeds

If you're sending the ping from a host in your on-premises network, and the ping succeeds,

that probably means your edge router is not honoring the Don't Fragment flag. Instead the router is fragmenting the large packet. The first fragment reaches the destination host, so the ping succeeds, which is misleading. If you try to do more than just ping, the fragments after the first get dropped, and the connection will hang.

Make sure to configure your router to honor the Don't Fragment flag. The router's default configuration is to honor it, but someone might have changed the default.

Bad: If you're testing the VCN side of the connection and you don't see the ICMP message

When testing from the VCN side of the connection, if you don't see the ICMP message in the response, there is probably something dropping the ICMP packet before it reaches your instance.

There could be two issues:

- **Security list:** The Networking security list could be missing an ingress rule that allows ICMP type 3 code 4 messages to reach the instance. This is an issue only if you're using stateless security list rules. If you're using stateful rules, your connections are tracked and the ICMP message is automatically allowed without needing a specific security list rule to allow it. **If you're using stateless rules, make sure that the subnet the instance is in has a security list with an ingress rule that allows ICMP traffic type 3 code 4 from source 0.0.0.0/0 and any source port.** For more information, see Security Lists, and specifically To update rules in an existing security list.
- **Instance firewall:** The instance's firewall rules (set in the OS) could be missing a rule that allows ICMP type 3 code 4 messages to reach the instance. Specifically for a Linux instance, make sure that iptables or firewalld is configured to allow the ICMP type 3 code 4 messages.

Avoiding the Need for PMTUD

Oracle recommends using PMTUD. However, in some situations it's possible to configure servers so they don't need to rely on it. Consider the case of the instances in your VCN communicating across an IPSec VPN to hosts in your on-premises network. You know the

range of IP addresses for your on-premises network. You can add a special route to your instances that specifies the maximum MTU to use when communicating with hosts in that address range. The instance-to-instance communication within the VCN still uses an MTU of 9000 bytes.

The following information shows how to set that route on a Linux instance.

The default route table on the instance typically has two routes: the default route (for the default gateway), and a local route (for the local subnet). For example:

```
ip route show
default via 10.0.6.1 dev ens3
10.0.6.0/27 dev ens3  proto kernel  scope link  src 10.0.6.9
```

You can add another route that points to the same default gateway, but with the address range of the on-premises network and a smaller MTU. For example, in the following command, the on-premises network is 1.0.0.0/8, the default gateway is 10.0.6.1, and the maximum MTU size is 1300 for packets being sent to the on-premises network.

```
ip route add 1.0.0.0/8 via 10.0.6.1 mtu 1300
```

The updated route table looks like this:

```
ip route show
default via 10.0.6.1 dev ens3
1.0.0.0/8 via 10.0.6.1 dev ens3  mtu 1300
10.0.6.0/27 dev ens3  proto kernel  scope link  src 10.0.6.9
```

Within the VCN, the instance-to-instance communication continues to use 9000 MTU. However, communication to the on-premises network uses a maximum of 1300. This example assumes there's no part of the connection between the on-premises network and VCN that uses an MTU smaller than 1300.



Important

The preceding commands do not persist if you reboot the instance. You can make the route permanent by adding it to a configuration file in the OS. For example, for Oracle Linux, it's an interface-specific file called `/etc/sysconfig/network-scripts/route-<interface>`. For more information, see the documentation for your variant of Linux.

Subnet or VCN Deletion

This topic covers reasons why deletion of a subnet or VCN might fail.

Remember:

- To delete a VCN, it must first be empty and have no related resources or attached gateways (for example: no internet gateway, dynamic routing gateway, and so on).
- To delete a VCN's subnets, they must first be empty.

The Subnet Isn't Empty

The most common reason a subnet (and thus a VCN) can't be deleted is because the subnet contains one or more of these resources:

- Mount target
- Load balancer
- DB system



Note

When you create one of the preceding resources, you specify a VCN and subnet for it, and a VNIC is automatically created in that subnet and attached to the resource. The VNIC enables the resource to communicate with other resources over the network. Although this documentation commonly talks about the resource itself being in the subnet, it's actually the resource's attached VNIC.

If the subnet *is* empty when you try to delete it, its state changes to TERMINATING briefly and then to TERMINATED. If it's not empty, you instead get an error indicating that there are still resources that you must delete first.

There Are Resources in Compartments You Don't Have Access To

You might not be able to see all the resources in a subnet or VCN. This is because subnets and VCNs can contain resources in multiple compartments, and you might not have access to all the compartments. For example, the subnet might contain instances that your team manages but also DB systems that another team manages. Another example: The VCN might have security lists or a gateway in a compartment that another team manages. You might need to contact your tenancy administrator to help you determine who owns the resources in the subnet or VCN.

CHAPTER 19 Notifications

This chapter explains how to use the Notifications service.

Notifications Overview

The Oracle Cloud Infrastructure Notifications service broadcasts messages to distributed components through a publish-subscribe pattern, delivering secure, highly reliable, low latency and durable messages for applications hosted on Oracle Cloud Infrastructure and externally. Use Notifications to get notified when alarms are breached. For more information about alarms, see Alarms Feature Overview.

How Notifications Works

The Notifications service enables you to set up communication channels for publishing messages using topics and subscriptions. When a message is published to a topic, the Notifications service sends the message to all of the topic's subscriptions.

When a subscriber’s endpoint does not acknowledge receipt of the message, the Notifications service retries delivery. This situation can occur when the endpoint is offline. For example, the email server for an email address may be down.

Delivery retry details

Notifications retries delivery following these steps until either (a) acknowledgement is received or (b) the subscription's retry duration is over. By default, the retry duration is two hours.

1. Immediate retry.
2. Exponential backoff retry for the period of the subscription's retry duration, using the following timing:

a. 1 minute
b. 2 minutes
c. 4 minutes
d. 8 minutes
e. 16 minutes
f. 32 minutes

3. Discarding of the message at the end of the retry duration.

You can change the retry duration for a subscription. For instructions using the Console, see To update the retry duration for a subscription (HTTPS (PagerDuty) protocol only). For the API, use the following operation: UpdateSubscription.

Notifications Concepts

The following concepts are essential to working with Notifications.

MESSAGE

The content that is published to a *topic*. Each message is delivered at least once per *subscription*. Every message sent out as email contains a link to *unsubscribe* from the related topic.

SUBSCRIPTION

An endpoint for a *topic*. Published *messages* are sent to each subscription for a *topic*. Supported subscription protocols include:

- Email - messages are sent to a specified email address.
- HTTPS (PagerDuty) - messages are sent to PagerDuty. For more information, see the PagerDuty documentation.

TOPIC

A communication channel for sending *messages* to the *subscriptions* in the topic.



Note

Messages sent out as email by the Oracle Cloud Infrastructure Notifications service are processed and delivered through Oracle resources in U.S.-based regions.

Availability

Notifications is currently available in the following regions:

Region Name	Region Location	Region Key
ap-tokyo-1	Asia-Pacific: Tokyo, Japan	NRT
ca-toronto-1	Canada: Toronto	YYZ
eu-frankfurt-1	Europe: Frankfurt, Germany	FRA
uk-london-1	United Kingdom: London	LHR
us-ashburn-1	United States: Ashburn, VA	IAD
us-phoenix-1	United States: Phoenix, AZ	PHX

Service Comparison for Sending Email Messages

Consider the following service features when deciding whether to use the Notifications service or the Email Delivery service to send your email messages. For more information about Email Delivery, see Overview of the Email Delivery Service.

Service Feature	Notifications service	Email Delivery service
Requires confirmation before sending email.	Yes	No
Allows email decorations, such as signatures.	Yes	No
Allows raw email messages.	No	Yes
Supports MIME attachments.	No	Yes
Supports special handling for failed email delivery.	No	Yes
Priced for small messages (less than 32KB, with a 64KB limit).	Yes	No
Priced for large messages (greater than 32KB, with a 2MB limit).	No	Yes

Resource Identifiers

Most types of Oracle Cloud Infrastructure resources have a unique, Oracle-assigned identifier called an Oracle Cloud ID (OCID). For information about the OCID format and other ways to identify your resources, see Resource Identifiers.

Ways to Access Notifications

You can access the Notifications service using the Console (a browser-based interface) or the REST API. Instructions for the Console and API are included in topics throughout this guide. For a list of available SDKs, see Software Development Kits and Command Line Interface.

Console: To access Notifications using the Console, you must use a supported browser. Oracle Cloud Infrastructure supports the latest desktop versions of Google Chrome, Microsoft Edge, Internet Explorer 11, Safari, Firefox, and Firefox ESR. Note that private browsing mode is not

supported for Firefox, Internet Explorer, or Edge. Mobile browsers are not supported. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Application Integration** and click **Notifications**.

API: To access Notifications through API, use Notifications API.

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

Administrators: For common policies that give groups access to Notifications, see Allow a group to manage topics, Allow a group to manage topic subscriptions, and Allow a group to publish messages to topics.

Limits on Notifications

See Notifications Limits for a list of applicable limits and instructions for requesting a limit increase.

Managing Topics and Subscriptions

This section describes how to manage topics and their subscriptions.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

A topic is a communication channel for sending messages to its subscriptions. A topic can have zero, one, or multiple subscriptions that are notified whenever a message is published to a topic.

Prerequisites

IAM policies: To use Notifications, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a response that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in. For more information on user authorizations, see Notifications Overview.

Using the Console

To create a topic

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Application Integration** and click **Notifications**.
2. Click **Create Topic** at the top of the topic list.

3. In the **Create Topic** dialog box, configure your topic.
    - **Name**: Required. Specify a friendly name for the topic. It must be unique; validation is case-sensitive. Avoid entering confidential information.
    - **Description**: Optional. Enter a description for the topic. Avoid entering confidential information.
4. Click **Create**.

To delete a topic

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Application Integration** and click **Notifications**.
2. For the topic you want to delete, click the Actions icon (three dots), and then click **Delete**.
3. Confirm when prompted.

To update the description for a topic

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Application Integration** and click **Notifications**.
2. Click the name of the topic you want to update.
3. On the topic detail page, next to **Description**, click the edit icon.
4. Edit the description.
5. Click the save icon.

To create a subscription

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Application Integration** and click **Notifications**.
2. Click the name of the topic that you want to add the subscription to.

3. On the topic detail page, click **Create Subscription**.
4. In the **Create Subscription** dialog box, configure your subscription for the protocol you want:

   ### Email subscription

   - **Protocol**: Select **Email**.
   - **Email**: Type an email address.

   ### PagerDuty subscription

   - **Protocol**: Select **HTTPS (PagerDuty)**.
   - **URL**: Type (or copy and paste) the *integration key* portion of the URL for your PagerDuty subscription. (The other portions of the URL are hard-coded.)
     For information on setting up and retrieving your integration key, see the PagerDuty documentation.

5. Click **Create**.
   The subscription has been created and a subscription confirmation URL will be sent. The subscription remains in "Pending" status until it has been confirmed.

To confirm a subscription

Steps depend on the subscription protocol.

- **Email** protocol: In the confirmation email sent to the specified email address, click the confirmation URL.
- **HTTPS (PagerDuty)** protocol: In PagerDuty, in the "Oracle Cloud Infrastructure Notifications Service Subscription Confirmation" incident, click the Confirmation URL in the message body.
  For more information, see the PagerDuty documentation for Oracle Cloud Infrastructure.

To resend a subscription confirmation

The ability to resend subscription confirmations is only applicable for pending subscriptions.

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Application Integration** and click **Notifications**.
2. For the subscription you want to resend the confirmation for, click the Actions icon (three dots), and then click **Resend Confirmation**.

To update the retry duration for a subscription (HTTPS (PagerDuty) protocol only)

You can edit the retry duration for subscriptions that use the **HTTPS (PagerDuty)** protocol. The retry duration is part of the delivery policy for the subscription. By default, Notifications will retry delivery of a message for up to two hours.

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Application Integration** and click **Notifications**.
2. For the subscription you want to update, click the Actions icon (three dots), and then click **Update Delivery Policy**.
   This option is available for subscriptions that use the **HTTPS (PagerDuty)** protocol only.
3. In the **Update Delivery Policy** dialog box, click the edit icon for **Max Retry Duration in Minutes**, type the new value, and then click the save icon.

To delete a subscription (unsubscribe)



Note

Every message sent out as email contains a link to *unsubscribe* from the related topic.

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Application Integration** and click **Notifications**.
2. For the subscription you want to delete, click the Actions icon (three dots), and then click **Delete**.
3. Confirm when prompted.

Managing Tags for a Topic or Subscription

You can apply tags to your resources, such as topics and subscriptions, to help you organize them according to your business needs. You can apply tags at the time you create a topic or subscription, or you can update the topic or subscription later with the desired tags. For general information about applying tags, see Resource Tags.

To manage tags for a topic

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Application Integration** and click **Notifications**.
2. Choose the **Compartment** that contains the topic you want to tag, and then click the topic's name.
3. Click the **Tags** tab to view or edit existing tags, or click **Apply Tag(s)** to add new ones.

For more information, see Resource Tags.

To manage tags for a subscription

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Application Integration** and click **Notifications**.
2. Choose the **Compartment** that contains the subscription you want to tag, and then click the name of the topic that has the subscription.

3. For the subscription you want to tag, click the Actions icon (three dots), and then click **Apply Tag(s)**.
   To view or edit existing tags, click the Actions icon (three dots), and then click **View Tags**.

For more information, see Resource Tags.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage topics:

- CreateTopic
- GetTopic
- ListTopics
- UpdateTopic
- DeleteTopic

Use these API operations to manage subscriptions:

- CreateSubscription
- GetSubscription
- ListSubscriptions
- UpdateSubscription
- GetConfirmSubscription
- ResendSubscriptionConfirmation
- GetUnsubscription
- DeleteSubscription

Publishing Messages

This topic describes how to publish messages using the Notifications service.

Each message is broadcast to all subscriptions in the specified topic. Every message sent out as email contains a link to *unsubscribe* from the related topic.

Message delivery rate limits per endpoint: 60 messages per minute for HTTPS (PagerDuty) protocol, 10 messages per minute for Email protocol.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Prerequisites

- IAM policies: To use Notifications, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a response that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in. For more information on user authorizations, see Notifications Overview.
- Before you can publish a message, you need a topic with at least one subscription. See Managing Topics and Subscriptions.

Using the Console

To publish a message

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Application Integration** and click **Notifications**.
2. On the **Topics** page, for the topic you want, click the Actions icon (three dots), and then click **Publish Message**.
3. In the **Publish Message** dialog box, fill in the fields:
   - **Title**: Enter the title you want to send.
     For the **Email** protocol, the title renders as the subject line of the email message. For the **HTTPS (PagerDuty)** protocol, the title renders as the title line of the message body within the PagerDuty notification.
   - **Message**: Enter the content you want to send.

     Note

     Message size limit per request: 64KB.
4. Click **Publish**.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to publish messages:

PublishMessage

Notifications Metrics

You can monitor the health, capacity, and performance of your messages by using metrics, alarms, and notifications.

This topic describes the metrics emitted by the metric namespace `oci_notification` (the Notifications service).

Resources: Not applicable. Measures data for messages, which are not resources.

Overview of the Notifications Service Metrics

The Notifications service metrics help you measure the number and size of messages that are in initial requests, are delivered, and are not delivered.

Prerequisites

- IAM policies: To monitor resources, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. The policy must give you access to the monitoring services as well as the resources being monitored. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in. For more information on user authorizations for monitoring, see the Authentication and Authorization section for the related service: Monitoring or Notifications.

Available Metrics

The metrics listed in the following table are automatically available for messages you publish to topics. You do not need to enable monitoring on any resources to get these metrics.

Each metric includes a subset of the following dimensions:

AVAILABILITYDOMAIN

The availability domain in which the associated topic resides.

ENDPOINTTYPE

The subscription protocol of the endpoint used for the delivery attempt.

REGION

The region in which the associated topic resides.

RESOURCEID

The OCID of the resource to which the metric applies.

TOPICDISPLAYNAME

The friendly name of the associated topic.

Metric	Metric Display Name	Unit	Description	Dimensions
`PublishedMessagesSize`	**Published Messages Size (Bytes)**	bytes	Size of messages in request.	`availabilityDomain` `region` `resourceId` `topicDisplayName`
`PublishedMessagesCount`	**Published Messages Count**	count	Count of messages in request.	

Metric	Metric Display Name	Unit	Description	Dimensions
`DeliveredMessagesSize`	**Delivered Messages Size (Bytes)**	bytes	Size of messages successfully delivered to end points.	`availabilityDomain` `endpointType` `region` `resourceId` `topicDisplayName`
`FailedMessagesSize`	**Failed Messages Sizes (Bytes)**	bytes	Size of messages that did not get delivered to end points.	
`DeliveredMessagesCount`	**Delivered Messages Count**	count	Count of messages successfully delivered to end points.	
`FailedMessagesCount`	**Failed Messages Count**	count	Count of messages that did not get delivered to end points.	

Using the Console

To view default metric charts for a single topic

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Application**

Integration and click **Notifications**.

2. Choose the **Compartment** that contains the topic you want to view, and then click the topic's name.
3. In the **Resources** menu, click **Metrics** (if necessary).
   The **Metrics** page displays a default set of charts for the current topic.

For more information about monitoring metrics and using alarms, see Monitoring Overview. For information about notifications for alarms, see Notifications Overview.

To view default metric charts for multiple topics

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Service Metrics**.
2. For **Metric Namespace**, select **oci_notification**.
   The **Service Metrics** page displays a default set of charts for the selected metric namespace. For more information about the emitted metrics, see the foregoing table. You can also use the Monitoring service to create custom queries.



Tip

- Filter metrics by dimension, such as a selected topic, by clicking **Add** above the charts (to the right of **Dimensions**).
- Aggregate data across all topics (show a single line in the chart) by selecting the check box for **Aggregate Metric Streams** on the right.

For more information about monitoring metrics and using alarms, see Monitoring Overview. For information about notifications for alarms, see Notifications Overview.

Using the API

Use the following APIs for monitoring:

- Monitoring API for metrics and alarms
- Notifications API for notifications (used with alarms)

CHAPTER 20 Object Storage

This chapter explains how to upload, manage, and access data using Object Storage.

Overview of Object Storage

Oracle Cloud Infrastructure offers two distinct storage class tiers to address the need for both performant, frequently accessed "hot" storage, and less frequently accessed "cold" storage. Storage tiers help you maximize performance where appropriate and minimize costs where possible.

- Use **Object Storage** for data to which you need fast, immediate, and frequent access. Data accessibility and performance justifies a higher price point to store data in the Object Storage tier.
- Use **Archive Storage** for data to which you seldom or rarely access, but that must be retained and preserved for long periods of time. The cost efficiency of the Archive Storage tier offsets the long lead time required to access the data. For more information, see Overview of Archive Storage.

About Object Storage

The Oracle Cloud Infrastructure Object Storage service is an internet-scale, high-performance storage platform that offers reliable and cost-efficient data durability. The Object Storage service can store an unlimited amount of unstructured data of any content type, including analytic data and rich content, like images and videos.

With Object Storage, you can safely and securely store or retrieve data directly from the internet or from within the cloud platform. Object Storage offers multiple management interfaces that let you easily manage storage at scale. The elasticity of the platform lets you start small and scale seamlessly, without experiencing any degradation in performance or service reliability.

Object Storage is a regional service and is not tied to any specific compute instance. You can access data from anywhere inside or outside the context of the Oracle Cloud Infrastructure, as

long you have internet connectivity and can access one of the Object Storage endpoints. Authorization and resource limits are discussed later in this topic.

Object Storage also supports private access from Oracle Cloud Infrastructure resources in a VCN through a *service gateway*. A service gateway allows connectivity to the Object Storage public endpoints from private IP addresses in private subnets. For example, you can back up DB systems to an Object Storage bucket over the Oracle Cloud Infrastructure backbone instead of over the internet. You can optionally use IAM policies to control which VCNs or ranges of IP addresses can access Object Storage. See Access to Oracle Services: Service Gateway for details.

The following list summarizes some of the ways that you can use Object Storage.

BIG DATA/HADOOP SUPPORT

You can use Object Storage as the primary data repository for big data. Object Storage offers a scalable storage platform that lets you store large datasets and operate seamlessly on those datasets. The HDFS connector provides connectivity to various big data analytic engines like Apache Spark and MapReduce. This connectivity enables the analytics engines to work directly with data stored in Object Storage. For more information, see Hadoop Support.

BACKUP/ARCHIVE

You can use Object Storage to preserve backup and archive data that must be stored for an extended duration to adhere to various compliance mandates.

CONTENT REPOSITORY

You can use Object Storage as your primary content repository for data, images, logs, and video. You can reliably store and preserve this data for a long time, and serve this content directly from Object Storage. The storage scales as your data storage needs scale.

LOG DATA

You can use Object Storage to preserve application log data so that you can retroactively analyze this data to determine usage pattern and debug issues.

LARGE DATASETS

You can use Object Storage to store generated application data that needs to be preserved for future use. Pharmaceutical trials data, genome data, and Internet of Things (IoT) data are examples of generated application data that you can preserve using Object Storage.

Object Storage Resources

The following summarizes the Object Storage resources. Authorization and resource limits are discussed later in this topic.

OBJECT

Any type of data, regardless of content type, is stored as an object. The object is composed of the object itself and metadata about the object. Each object is stored in a bucket.

BUCKET

A logical container for storing objects. Users or systems create buckets as needed within a region. A bucket is associated with a single compartment that has policies that determine what actions a user can perform on a bucket and on all the objects in the bucket.

NAMESPACE

A logical entity that serves as a top-level container for all buckets and objects, allowing you to control bucket naming within your tenancy. Each tenancy is provided one unique and uneditable Object Storage namespace that is global, spanning all compartments and regions. Bucket names must be unique within your tenancy, but the use of a bucket name by another tenant does not restrict your ability to use the same bucket name within your tenancy. Within an Object Storage namespace, buckets and objects exist in flat hierarchy, but you can simulate a directory structure to help navigate a large set of objects (for example, `guitars/fender/stratocaster.jpg, guitars/gibson/lespaul.jpg`).



Tip

If your namespace was created based on your tenancy name, your namespace uses all lower-case letters (regardless of the presence of capital letters in your tenancy name). When using the API, CLI, or SDKs, do not use capital letters in your namespace string.

COMPARTMENT

Primary building block used to organize your cloud resources. When your tenancy is provisioned, a root compartment is created for you. You can then create compartments under your root compartment to organize your resources. You control access by creating policies that specify what actions groups of users can take on the resources in those compartments. An Object Storage bucket can only exist in one compartment.

Object Storage Features

Object Storage provides the following features:

STRONG CONSISTENCY

When a read request is made, Object Storage always serves the most recent copy of the data that was written to the system.

DURABILITY

Object Storage is a regional service. Data is stored redundantly across multiple storage servers and across multiple availability domains. Object Storage actively monitors data integrity using checksums and automatically detects and repairs corrupt data. Object Storage actively monitors and ensures data redundancy. If a redundancy loss is detected, Object Storage automatically creates more data copies.

CUSTOM METADATA

You can define your own extensive metadata as key-value pairs for any purpose. For example, you can create descriptive tags for objects, retrieve those tags, and sort through the data. You can assign custom metadata to objects and buckets using the Oracle Cloud Infrastructure CLI or SDK. See Software Development Kits and Command Line Interface for details.

ENCRYPTION

Object Storage employs 256-bit Advanced Encryption Standard (AES-256) to encrypt object data on the server. Each object is encrypted with its own key. Object keys are encrypted with a master encryption key that is frequently rotated. Encryption is enabled by default and cannot be turned off.

Ways to Access Object Storage

You can access Object Storage using any of the following options, based on your preference and its suitability for the task you want to complete:

- The Console is an easy-to-use, browser-based interface. To access the Console, you must use a supported browser. Oracle Cloud Infrastructure supports the latest desktop versions of Google Chrome, Microsoft Edge, Internet Explorer 11, Safari, Firefox, and Firefox ESR. Note that private browsing mode is not supported for Firefox, Internet Explorer, or Edge. Mobile browsers are not supported. You are prompted to enter your cloud tenant, your user name, and your password.
- The command line interface (CLI) provides both quick access and full functionality without the need for programming. For more information, see Using the CLI.
- The REST API provides the most functionality, but requires programming expertise. API Reference and Endpoints provides endpoint details and links to the available API reference documents. For general information about using the API, see REST APIs.
- The Oracle Cloud Infrastructure SDKs offer tools to interact with Object Storage without having to create a framework. For general information about using the SDKs, see Software Development Kits and Command Line Interface.

Using Object Storage

If you are ready to use Object Storage, you can find more information in the following topics:

- For instructions on how to create a bucket and store an object in the bucket, see "Putting Data into Object Storage" in the *Oracle Cloud Infrastructure Getting Started Guide*.
- For task documentation related to buckets, see Managing Buckets.
- For task documentation related to objects, see Managing Objects and Copying Objects.
- For task documentation related to lifecycle management, see Using Object Lifecycle Management
- For API reference documentation, see Object Storage Service API.
- For SDK and CLI information, see Software Development Kits and Command Line Interface.
- For more information about using Archive Storage, see Overview of Archive Storage.

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

Object Storage IP Addresses

The Oracle Cloud Infrastructure Object Storage service uses the CIDR block IP range 134.70.0.0/17 for all regions.

Limits on Object Storage Resources

See Service Limits for a list of applicable limits and instructions for requesting a limit increase.

Other limits include:

- Number of Object Storage namespaces per root compartment: 1
- Maximum size of object metadata: 2 K

Understanding Object Storage Namespaces

Each Oracle Cloud Infrastructure tenant is assigned one unique and uneditable Object Storage namespace that spans all compartments within a region. The Object Storage namespace serves as a top-level container for all buckets and objects and lets you control bucket naming within your tenancy. Bucket names must be unique within the context of a namespace, but bucket names can be repeated across namespaces or across regions.

The Object Storage namespace is a system-generated string assigned during account creation. Note that for some older tenancies, the namespace string may be the tenancy name in all lower-case letters.

The namespace metadata stores the default compartment assignments for the Amazon S3 Compatibility API and the Swift API. For more information, see Viewing and Specifying Designated Compartments.

Using the Console

To view your Object Storage namespace string:

Open the **User** menu () and click **Tenancy:** *<your_tenancy_name>*. Your namespace string is listed under **Object Storage Settings**.



Note

Object Storage Namespace Is Not Editable

While the Object Storage namespace string is displayed under **Object Storage Settings**, you cannot edit the namespace string. The namespace string appears here for information only.

Using the Command Line Interface (CLI)

Open a command prompt and run the following command to get your Object Storage namespace:

```
oci os ns get
```



Tip

You can use `-ns`, `--namespace`, or `--namespace-name` for options that require you to specify the Object Storage namespace string.

For information about using the CLI, see Command Line Interface (CLI). For a complete list of flags and options available for CLI commands, see CLI Help.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the GetNamespace operation to get your Object Storage namespace. You can also get the namespace of a different tenancy's Object Storage namespace if you have the OBJECTSTORAGE_NAMESPACE_READ permission and supply the compartment or tenancy OCID in the optional `compartmentId` parameter.

Managing Buckets

In the Oracle Cloud Infrastructure Object Storage service, a bucket is a container for storing objects in a compartment within an Object Storage namespace. A bucket is associated with a single compartment. The compartment has policies that indicate what actions a user can perform on a bucket and all the objects in the bucket.

You cannot nest buckets—a bucket cannot contain other buckets.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

If you're new to policies, see Getting Started with Policies and Common Policies.

For administrators:

- The policy Let Object Storage admins manage buckets and objects lets the specified group do everything with buckets and their associated objects.
- If you need to write more restrictive policies for buckets, see Details for Object Storage, Archive Storage, and Data Transfer.

Pre-Authenticated Requests

Pre-authenticated requests provide a way to let users access a bucket or an object without having their own credentials. For example, you can create a request that lets a user upload

backups to a bucket without owning API keys. See Using Pre-Authenticated Requests for details.

Object Lifecycle Policies

Using object lifecycle policies applied at the bucket level, you can automatically manage the archiving and deletion of objects according to a pre-defined schedule. See Using Object Lifecycle Management for information on this feature.

Tagging Resources

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

Object Storage currently supports applying tags to buckets.

Monitoring Resources

You can monitor the health, capacity, and performance of your Oracle Cloud Infrastructure resources by using metrics, alarms, and notifications. For more information, see Monitoring Overview and Notifications Overview.

For more information about monitoring buckets, see Object Storage Metrics.

Bucket Names

Bucket names are system generated by default, but you can overwrite the default with a name you specify.

System-Generated Bucket Names

When a bucket is created, the system generates a default name for that bucket, for example **bucket-20190306-1359**. This bucket name identifies the current year, month, and day that

the bucket was created. You can use that system-generated name for your new bucket or you can specify a different name for it.

User-Specified Bucket Names

If you change this default bucket name or the name of any bucket, observe the following:

- Use from 1 to 256 characters.
- Valid characters are letters (upper or lower case), numbers, hyphens, underscores, and periods.



Important

Bucket names and object names are case-sensitive. Object Storage handles accounts-payable and Accounts-Payable as separate buckets.

- Do not include confidential information.
- Make the name unique within your tenancy's Object Storage namespace.

Storage Tiers

When you create a bucket, you also decide which tier is appropriate for storing objects:

- Use the standard Object Storage tier for data to which you need fast, immediate, and frequent access. For more information, see Overview of Object Storage.
- Use the Archive Storage tier for data to which you seldom or rarely access, but that must be retained and preserved for long periods of time. For more information, see Overview of Archive Storage.



Important

Once set, you cannot change the storage tier in which a bucket resides.

Public Buckets

When you create a bucket, the bucket is considered a private bucket and the access to the bucket and its contents requires authentication and authorization. However, Object Storage supports anonymous, unauthenticated access to a bucket. You make a bucket *public* by enabling read access to the bucket.



Important

Carefully assess the business requirement for public access to a bucket. When you enable anonymous access to a bucket, users can obtain object metadata, download bucket objects, and optionally list bucket contents.

Required Permissions

The following permissions are required to configure a public bucket:

- To enable public access when creating a bucket, use permission `BUCKET_CREATE`.
- To enable public access for an existing bucket, use permission `BUCKET_UPDATE`.

Options

When creating a public bucket, you have the following options:

- You can configure the access to allow listing and downloading objects. List and download access is the default.
- You can configure the access to allow downloading objects only. Users would not be able to list bucket contents.

Scope and Constraints

Understand the following scope and constraints regarding public access:

- Changing the type of access is bi-directional. You can change a bucket's access from public to private or from private to public.
- Changing the type of access doesn't affect existing pre-authenticated requests. Existing pre-authenticated requests still work.

You can enable anonymous public access for new or existing buckets using the Console, CLI, or an SDK to access the API.

Using the Console

To get a list of buckets

Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.

A list of the buckets in the compartment you're viewing is displayed. If you don’t see the one you're looking for, verify that you’re viewing the correct compartment (select from the list on the left side of the page).

To create a bucket

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
   A list of the buckets in the compartment you're viewing is displayed. If you don’t see the one you're looking for, verify that you’re viewing the correct compartment (select from the list on the left side of the page).

2. Choose the compartment for your new bucket.
   A list of existing buckets is displayed.
3. Click **Create Bucket**.
4. In the **Create Bucket** dialog box, specify the attributes of the bucket:
   - **Bucket Name**: The system generates a default bucket name that reflects the current year, month, day, and time, for example **bucket-20190306-1359**. If you change this default to any other bucket name, use letters, numbers, dashes, underscores, and periods. Do not include any confidential information.
   - **Storage Tier**: Select the tier in which you want to store your data. Available tiers include:
     - **Standard** is the primary, default Object Storage tier for storing frequently accessed data that requires fast and immediate access.
     - **Archive** is a special tier for storing infrequently accessed data that requires long retention periods. Access to data in the **Archive** tier is not immediate. You must restore archived data before it's accessible. For more information, see "Overview of Archive Storage" in the *Oracle Cloud Infrastructure User Guide.*
   - **Encryption:** Buckets are encrypted with keys managed by Oracle by default, but you can optionally encrypt the data in this bucket using your own Key Management encryption key. To use Key Management for your encryption needs, select **Encrypt Using Customer-Managed Keys**. Then, select the **Vault Compartment** and **Vault** that contain the master encryption key you want to use. Also select the **Master Encryption Key Compartment** and **Master Encryption Key**. For more information about encryption, see Overview of Key Management. For details on how to create a vault, see Managing Vaults.
   - **Tags:** Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you

should apply tags, skip this option (you can apply tags later) or ask your administrator.

5. Click **Create Bucket**.

The bucket is created immediately and you can add objects to it. Objects added to archive buckets are immediately archived and must be restored before they are available for download.

To view bucket details

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
2. Choose the compartment that contains your buckets.
   A list of buckets is displayed.
3. Click the Actions icon (three dots) to the right of the bucket name, and then click **View Bucket Details**.
   Object Storage displays bucket details including the following:
   - Visibility
   - Encryption Key
   - Namespace
   - Created
   - Storage tier
   - Compartment
   - Approximate Count
   - Approximate Size
   - ETag (entity tag)

To view the approximate bucket size and number of objects in the bucket

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.

2. Choose the compartment that contains your buckets.
   A list of buckets is displayed.
3. Click the Actions icon (three dots) to the right of the bucket name, and then click **View Bucket Details**.
   - **Approximate Count** is the approximate number of objects in the bucket. Count statistics are reported periodically. You will see a lag between what is displayed and the actual object count.
   - **Approximate Size** is the approximate total size of all objects in the bucket. Size statistics are reported periodically. You will see a lag between what is displayed and the actual size of the bucket.

To move a bucket to a different compartment

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
2. Choose the compartment containing the bucket that you want to move.
3. Click the Actions icon (three dots), and then click **Move Bucket**. Alternatively, you can choose a bucket, and then click **Move Bucket** on the **Bucket Details** page.
4. In the **Change Compartment** dialog box, specify the target compartment that you want to move the bucket to and click **Move the bucket to the selected compartment**.

To delete a bucket

You can permanently delete an *empty* bucket. The bucket cannot contain any objects. For information on deleting objects from a bucket, see To delete objects from a bucket. In addition, you cannot delete a bucket that has a multipart upload in progress or a pre-authenticated request associated with that bucket.



Warning

You cannot recover a deleted bucket.

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
   A list of the buckets in the compartment you're viewing is displayed. If you don't see the one you're looking for, verify that you're viewing the correct compartment (select from the list on the left side of the page).
2. Find the bucket that you want to delete.
3. Click the Actions icon (three dots), and then click **Delete**.
4. Confirm deletion when prompted.

To manage tags for a bucket

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
   A list of the buckets in the compartment you're viewing is displayed. If you don't see the one you're looking for, verify that you're viewing the correct compartment (select from the list on the left side of the page).
2. Click the bucket name.
3. To view existing tags, click the **Tags** tab.
4. Optionally, rename or remove tags associated with the bucket by doing one of the following:
   - Click the pencil icon to the left of a tag name to edit and save its tag key name.
   - Click the pencil icon to the left of a tag name and click **Remove Tag** to delete the tag.
5. Optionally, click the **Apply tags** button to add one or more tags.
   For each new tag, specify a Tag namespace (required), a Tag key (required), and a Value (optional).

For more information, see Resource Tags.

To change the visibility of a bucket

A bucket is either private (the default) or public. See Public Buckets for more information.

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
   A list of the buckets in the compartment you're viewing is displayed. If you don't see the one you're looking for, verify that you're viewing the correct compartment (select from the list on the left side of the page).
2. Click the bucket name to see the bucket details.
   Visibility: shows the current bucket setting, which is **Private** by default.
3. Click **Edit Visibility**.
4. In the **Edit Visibility** dialog box, edit the visibility settings:
   - **Visibility**
     - **Public**
     - **Private**
   - If you select **Public** to enable public access, decide whether or not you want to let users list the bucket contents. Click **Allow users to list objects from this bucket** to set the visibility of bucket object lists.
5. Click **Save Changes**.

To assign or edit a bucket Key Management key to a bucket

Buckets are encrypted with keys managed by Oracle by default, but you can optionally encrypt the data in this bucket using your own Key Management encryption key.

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.

A list of the buckets in the compartment you're viewing is displayed. If you don't see the one you're looking for, verify that you're viewing the correct compartment (select from the list on the left side of the page).

2. Click the name of the bucket that you want to encrypt.
3. Next to **Encryption Key**, do one of the following:
   - If the bucket does not have a key already, click the **Assign** link.
   - If the bucket already has a key assigned, to assign a different key, click the **Edit** link.
4. In the dialog box, provide or edit the following information:
   - **Vault Compartment** and **Vault** that contain the master encryption key you want to use. The system current compartment is displayed by default.
   - **Master Encryption Key Compartment** and **Master Encryption Key**. The system current compartment is displayed by default.
5. When you are finished, click **Assign** or **Edit**.

See Overview of Key Management for more details.

To remove a Key Management key from a bucket

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
   A list of the buckets in the compartment you're viewing is displayed. If you don't see the one you're looking for, verify that you're viewing the correct compartment (select from the list on the left side of the page).
2. Click the name of the bucket for which you want to remove a Key Management key assignment.
3. Next to **Encryption Key**, click the **Unassign** link.
4. In the **Confirm** dialog box, click **OK** to remove the key assignment from the bucket.

Using the Command Line Interface (CLI)

For information about using the CLI, see Command Line Interface (CLI). For a complete list of flags and options available for CLI commands, see CLI Help.

To get a list of buckets

Open a command prompt and run `oci os bucket list` to list the buckets in a compartment.

To list the first 1,000 buckets in a compartment:

```
oci os bucket list -ns <object_storage_namespace> --compartment-id <target_compartment_id>
```

To list all of the buckets in a compartment, use the `--all` option:

```
oci os bucket list -ns <object_storage_namespace> --compartment-id <target_compartment_id> --all
```

For example:

```
oci os bucket list -ns ansh8tvru7zp -c ocid1.compartment.oc1..<unique_ID>

{
  "data": [
    {
                "compartment-id": "ocid1.compartment.oc1..exampleuniqueID",
                "created-by": "ocid1.user.oc1..exampleuniqueID",
                "defined-tags": null,
                "etag": "3f06cb1d-2a95-40e3-afa8-example28553",
                "freeform-tags": null,
                "name": "example_bucket_name",
                "namespace": "ansh8tvru7zp",
                "time-created": "2018-02-15T23:32:44.147000+00:00"
        }
  ]
}
```

To include resource tag data, use the `--fields tags` option:

```
oci os bucket list -ns <object_storage_namespace> --compartment-id <target_compartment_id> --fields
tags
```

For example:

```
oci os bucket list -ns ansh8tvru7zp -c ocid1.compartment.oc1..<unique_ID> --fields tags

{
  "data": [
       {
         "compartment-id": "ocid1.compartment.oc1..exampleuniqueID",
         "created-by": "ocid1.user.oc1..exampleuniqueID",
         "defined-tags": {
               "example_tag_namespace_Financials": {
                 "production": "Unit 5"
               },
               "example_tag_namespace_Operations": {
                 "costcenter": "85"
               }
         },
         "etag": "48af18cf-1edd-4b05-9f36-a629d5032260",
         "freeform-tags": {
           "Project": "prototype 3"
         },
         "name": "example_bucket_name",
         "namespace": "ansh8tvru7zp",
         "time-created": "2018-02-27T18:52:16.951000+00:00"
       }
  ]
}
```



Note

If you do not specify the `--fields tags` option when listing buckets, `null` is returned as the value for both freeform and defined tags.

To create a standard Object Storage tier bucket

Open a command prompt and run `oci os bucket create` to create a bucket. Avoid entering confidential information in bucket name.

```
oci os bucket create -ns <object_storage_namespace> --name <bucket_name> --compartment-id <target_compartment_id>
```

By default, a bucket is created in the standard Object Storage tier. You can set `--storage-tier` explicitly, though it is not required.

A Standard tier bucket is created immediately and you can add objects to it.

To create an Archive tier bucket

Open a command prompt and run `oci os bucket create` to create a bucket. Avoid entering confidential information in bucket name.

To create an Archive tier bucket, you must explicitly set `--storage-tier`. For example:

```
oci os bucket create -ns <object_storage_namespace> --name <archivebucket_name> --compartment-id <target_compartment_id> --storage-tier Archive
```

An Archive Storage bucket is created and you can add objects to it. Objects added to Archive Storage buckets are immediately archived and must be restored before they are available for download.

To view bucket details

Open a command prompt and run `oci os bucket get` to get bucket details.

```
oci os bucket get --name <bucket_name>
```

For example:

```
oci os bucket get --name MyBucket
{
  "data": {
    "compartment-id": "ocid1.compartment.oc1..exampleuniqueID",
    "created-by": "ocid1:user:oc1:phx:1458751937789:exampleuniqueID",
```

```
    "defined-tags": {},
    "estimated-count": null,
    "estimated-size": null,
    "etag": "218f201f-28a4-434d-9591-f05b6223c67a",
    "freeform-tags": {},
    "kms-key-id": null,
    "metadata": {},
    "name": "MyBucket",
    "namespace": "docs",
    "object-level-audit-mode": "Disabled",
    "object-lifecycle-policy-etag": null,
    "public-access-type": "NoPublicAccess",
    "storage-tier": "Standard",
    "time-created": "2017-10-19T04:11:32.040000+00:00"
  },
  "etag": "218f201f-28a4-434d-9591-f05b6223c67a"
}
```

To view the approximate bucket size and number of objects in the bucket

```
oci os bucket get --name <bucket_name> --fields approximateCount --fields approximateSize
```

- approximateCount is the approximate number of objects in the bucket. Count statistics are reported periodically. You will see a lag between what is displayed and the actual object count.
- approximateSize is the approximate total size of all objects in the bucket. Size statistics are reported periodically. You will see a lag between what is displayed and the actual size of the bucket.

For example:

```
oci os bucket get --name MyBucket --fields approximateCount --fields approximateSize
{
  "data": {
    "compartment-id": "ocid1.compartment.oc1..exampleuniqueID",
    "created-by": "ocid1:user:oc1:phx:1458751937789:exampleuniqueID",
    "defined-tags": {},
    "approximate-count": 7,
    "approximate-size": 8075918,
```

```
    "etag": "218f201f-28a4-434d-9591-f05b6223c67a",
    "freeform-tags": {},
    "kms-key-id": null,
    "metadata": {},
    "name": "MyBucket",
    "namespace": "docs",
    "object-level-audit-mode": "Disabled",
    "object-lifecycle-policy-etag": null,
    "public-access-type": "NoPublicAccess",
    "storage-tier": "Standard",
    "time-created": "2017-10-19T04:11:32.040000+00:00"
  },
  "etag": "218f201f-28a4-434d-9591-f05b6223c67a"
}
```

To make a bucket private or public

```
oci os bucket update -ns <object_storage_namespace> --name <bucket_name> --public-access-type
<visibility_setting>
```

Where *<visibility_setting>* is "NoPublicAccess", "ObjectReadWithoutList", or "ObjectRead" and represents the new setting for the bucket.

To create a public bucket that allows listing and downloading bucket objects

```
oci os bucket create -ns <object_storage_namespace> --name <bucket_name> --public-access-type
ObjectRead -c <compartment_ocid>
```

Where *<compartment_ocid>* is a value like: `ocid1.compartment.oc1..exampleuniqueID`

Avoid entering confidential information in bucket name.

To create a public bucket that allows downloading bucket objects only

```
oci os bucket create -ns <object_storage_namespace> --name <bucket_name> --public-access-type
ObjectReadWithoutList -c <compartment_ocid>
```

Where *<compartment_ocid>* is a value like: `ocid1.compartment.oc1..exampleuniqueID`

Avoid entering confidential information in bucket name.

To create a bucket with resource tags

You can create standard Object Storage tier or Archive tier buckets with resource tags.

To add resource tags when creating a bucket, open a command prompt and run `oci os bucket create` with one or both of the `--defined-tags` and `--freeform-tags` options.

The following example syntax creates a standard Object Storage tier bucket with a defined tag:

```
oci os bucket create -ns <object_storage_namespace> --name <bucket_name> --compartment-id <target_compartment_id> --defined-tags <JSON_formatted_defined_tag>
```

The following example syntax creates a Standard tier bucket with a free-form tag:

```
oci os bucket create -ns <object_storage_namespace> --name <bucket_name> --compartment-id <target_compartment_id> --freeform-tags <JSON_formatted_free-form_tag>
```



Tip

The `--defined-tags` and `--freeform-tags` options require that the input be a complex type formatted in valid JSON. See Passing Complex Input and Using a JSON File for Complex Input for information JSON formatting.

Examples of defined tag formatting:

```
'{"Logistics": {"Procurement": "Madrid Center"}}'
```

```
'{"Operations": {"CostCenter": "42"},"Financials":{"Production": "Unit 5"}}'
```

Examples of free-form tag formatting:

```
'{"Chicago Team": "marketing videos"}'
```

```
'{"Project": "prototype 3","Manager": "Meadows"}'
```

If you are using a Windows CLI, you may need to use the backslash (\) character to escape the strings containing the tag values. For example, a single defined tag is formatted as follows:

```
'{\"Logistics\": {\"Procurement\": \"Madrid Center\"}}'
```



Warning

Avoid entering confidential information in bucket name.

To move a bucket to a different compartment

Open a command prompt and run `oci os bucket update` to move a bucket to a different compartment.

```
oci os bucket update -ns <object_storage_namespace> --name <bucket_name> --compartment-id <new_target_compartment_id>
```

To delete a bucket

You can permanently delete an *empty* bucket. The bucket cannot contain any objects. For information on deleting objects, see To delete objects from a bucket. You also cannot delete a bucket that has a multipart upload in progress or a pre-authenticated request associated with that bucket.

Open a command prompt and run `oci os bucket delete` to delete a bucket. For example:

```
oci os bucket delete -ns <object_storage_namespace> --name <bucket_name>
```



Warning

You cannot recover a deleted bucket.

To get bucket metadata

You can get bucket metadata using the CLI. This metadata includes the information that is displayed in the Console **Bucket Details**.

Open a command prompt and run `oci os bucket get` to get information about an individual bucket. For example:

```
oci os bucket get -ns <object_storage_namespace> --name <bucket_name>
```

The command returns the following information:

- Bucket visibility (private or public)
- Bucket name
- Compartment ID
- Entity tag (ETag)
- Custom metadata key-value pairs
- Namespace of the bucket
- Storage tier of the bucket
- Timestamp for bucket creation
- User OCID for the bucket creator

To add custom metadata key-value pairs to a bucket

Open a command prompt and run `oci os bucket update` with the `--metadata` option to add key-value pairs to a bucket:

```
oci os bucket update -ns <object_storage_namespace> --name <bucket_name> --metadata <json_formatted_
key-value_pairs>
```



Tip

The `--metadata` option requires that you provide key-value pair input as valid formatted JSON. See Passing Complex Input and Using a JSON File for Complex Input for more information about JSON formatting.

To add resource tags to a bucket

To add defined resource tags to a bucket, open a command prompt and run `oci os bucket update` with the `--defined-tags` option:

```
oci os bucket update -ns <object_storage_namespace> --name <bucket_name> --defined-tags <JSON_formatted_defined_tag>
```

To add free-form resource tags to a bucket, open a command prompt and run `oci os bucket update` with the `--freeform-tags` option:

```
oci os bucket update -ns <object_storage_namespace> --name <bucket_name> --freeform-tags <JSON_formatted_free-form_tag>
```



Tip

The `--defined-tags` and `--freeform-tags` options require that you provide key-value pair input as valid formatted JSON. For examples of JSON-formatted resource tags, see To create a Standard or Archive tier bucket with resource tags. See Passing Complex Input and Using a JSON File for Complex Input for more information about JSON formatting.

To assign a Key Management key to an Object Storage bucket

Open a command prompt and run `oci os bucket create` to create a bucket that is encrypted with a Key Management master encryption key. Do not enter confidential information in the bucket name.

```
oci os bucket create --name <bucket_name> --compartment-id <target_compartment_id> --kms-key-id
<target_key_id>
```

For example:

```
oci os bucket create --name Bucket-1 --compartment-id
ocid1.compartment.oc1..ocid1.compartment.oc1..exampleuniqueID --kms-key-id
ocid1.key.region1.sea.exampleuniqueID --namespace-name example_namespace
```

See Overview of Key Management for more details.

To update the Key Management key assigned to an Object Storage bucket

Open a command prompt and run `oci os bucket update` to update the Key Management master encryption key assigned to a bucket:

```
oci os bucket update --name <bucket_name> --namespace-name <your_namespace> --kms-key-id <target_key_
id>
```

For example:

```
oci os bucket update --name Bucket-1 --namespace-name example_namespace --kms-key-id
ocid1.key.region1.sea.exampleuniqueID
```

To remove the Key Management key assigned to an Object Storage bucket

Open a command prompt and run `oci os bucket update` to remove the Key Management master encryption key assigned to a bucket:

```
oci os bucket update --name <bucket_name> --namespace-name <your_namespace> --kms-key-id ""
```

For example:

```
oci os bucket update --name Bucket-1 --kms-key-id "" --namespace-name example_namespace
```

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

When accessing the Object Storage API, the bucket name is used with the Object Storage namespace name to form the request URL:

```
n/<object_storage_namespace>/b/<bucket>
```

Use the following operations to manage buckets:

- CreateBucket (The **publicAccessType** property controls whether the bucket is private or public and limits the capability to list public bucket contents.)
- DeleteBucket
- GetBucket
- HeadBucket
- ListBuckets
- UpdateBucket

Managing Objects

In the Oracle Cloud Infrastructure Object Storage service, an object is a file or unstructured data you upload to a bucket within a compartment within an Object Storage namespace. The object can be any type of data, for example, multimedia files, data backups, static web content, or logs. You can store objects that are up to 10 TiB. Objects are processed as a single entity. You can't edit or append data to an object, but you can replace the entire object.

This topic describes how to manage objects within a single bucket. For information on copying an object to another bucket, see Copying Objects.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

If you're new to policies, see Getting Started with Policies and Common Policies.

For administrators:

- The policy Let Object Storage admins manage buckets and objects lets the specified group do everything with buckets and objects. Objects always reside in the same compartment as the bucket.
- If you need to write a more restrictive policy for objects, the `inspect objects` lets you list all the objects in a bucket and do a HEAD operation for a particular object. In comparison, `read objects` lets you download the object itself. See Details for Object Storage, Archive Storage, and Data Transfer.

Pre-Authenticated Requests

Pre-authenticated requests provide a way to let users access a bucket or object without having their own credentials. For example, you can create a request that lets a user upload backups to a bucket without owning API keys. See Using Pre-Authenticated Requests for details.

Object Names

Unlike other resources, objects do not have Oracle Cloud Identifiers (OCIDs). Instead, users define an object name when they upload an object.

Use the following guidelines when naming an object:

- Use from 1 to 1024 characters.
- Valid characters are letters (upper or lower case), numbers, and characters other than linefeed, newline, and NULL.



Important

Bucket names and object names are case-sensitive. Object Storage handles q3-field-assets.xslx and Q3-Field-Assets.XSLX as separate objects.

- Use only Unicode characters for which the UTF-8 encoding does not exceed 1024 bytes. Clients are responsible for URL-encoding characters.
- Do not include confidential information.
- Make the name unique within the bucket. Do not use the name of an existing object within the bucket when naming an object unless you intend to overwrite the existing object with the contents of the new or renamed object.



Tip

Object names can include one or more forward slash (/) characters in the name . See Object Naming Using Prefixes and Hierarchies for more information on using the forward slash in object names to create hierarchies.

Object Naming Using Prefixes and Hierarchies

Within an Object Storage namespace, buckets and objects exist in a flat hierarchy, but you can simulate a directory structure using a prefix string that includes the forward slash (/) to add hierarchy to an object name. Doing so lets you list one directory at a time, which is helpful when navigating a large set of objects.

For example:

```
marathon/finish_line.jpg
marathon/participants/p_21.jpg
```

If you named your objects so that they exist in Object Storage as a hierarchy, you can use the command line interface (CLI) or API to perform bulk downloads and bulk deletes of all objects at a specified level of the hierarchy, without affecting objects in levels above.

When naming objects, you can also use prefix strings without a delimiter so that certain bulk operations can be performed in the CL or API by matching on the prefix portion of the object name. For example, in the object names below, the string `gloves_27_` can serve as a prefix for matching purposes when performing bulk downloads or deletions:

```
gloves_27_dark_green.jpg
gloves_27_light_blue.jpg
```

When you perform bulk uploads with the CLI or API, you can also prepend a prefix string to the names of the files you are uploading.

Object Lifecycle Management

Using Object Lifecycle Management feature, you can automatically manage the archiving and deletion of objects according to a pre-defined schedule. See Using Object Lifecycle Management for information on this feature.

Multipart Uploading and Downloading

The Oracle Cloud Infrastructure Object Storage service supports multipart uploading and downloading for objects. See Using Multipart Uploads for more information. This page includes links to API documentation for this functionality. For information on multipart downloading, see the CLI procedure To download an object using multipart download. For API documentation related to multipart downloading, see the GetObject API call and its **range** parameter.

Monitoring Resources

You can monitor the health, capacity, and performance of your Oracle Cloud Infrastructure resources by using metrics, alarms, and notifications. For more information, see Monitoring Overview and Notifications Overview.

For more information about monitoring objects, see Object Storage Metrics.

Using Storage Gateway to Upload and Download Objects

Storage Gateway is another way you can upload objects to and download objects from Oracle Cloud Infrastructure Object Storage.

Storage Gateway is installed in an Oracle Cloud Infrastructure compute instance or as a Linux Docker instance on one or more hosts in your on-premises data center. Applications store and retrieve objects from Oracle Cloud Infrastructure Object Storage through *file systems* that you create in Storage Gateway. Storage Gateway exposes an NFS mount point that can be mounted to any host that supports an NFSv4 client. The Storage Gateway mount point maps to an Object Storage bucket to upload and download objects.

See Overview of Storage Gateway for details.

Using the Console

To upload objects to a bucket

1. From the Object Storage Details screen, click the bucket name to view its details.
2. Click **Objects** under **Resources**.
3. In the **Objects** table, click **Upload Objects**.
4. Optionally, specify an **Object Name Prefix**. If provided, this prefix is prepended to each one of the files you upload. The prefix lets you simulate hierarchy and perform bulk operations. See Object Naming Using Prefixes and Hierarchies for details.

5. In the **Upload Objects** dialog box, select the objects that you want to upload in one of two ways:
   - Drag and drop one or more files from your computer.
   - Click the **select files** link and use the **File Upload** dialog box.

   The files you select to upload are displayed in a list. If you decide that you do not want to upload a particular file, click the **X** to the right of the file name.
   If the files you select to upload are already stored in the bucket with the same name, the Console displays messages warning you of an overwrite.
6. Click **Upload Objects**.
   The selected objects are uploaded and displayed in the list of objects in the bucket.

To download an object from a bucket

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
2. Choose the compartment that contains the bucket that contains your object.
   A list of buckets is displayed.
3. Click the bucket name that contains your object.
4. Click **Objects** under **Resources**.
   A list of objects in the bucket is displayed.
5. For the object you want to download, click the Actions icon (three dots), and then click **Download**.

To view object details

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
2. Choose the compartment that contains the bucket that contains your object.
   A list of buckets is displayed.
3. Click the bucket name that contains your object.

4. Click **Objects** under **Resources**.
   A list of objects in the bucket is displayed.
5. Choose the object for which you want details.
6. Click the Actions icon (three dots), and then click **View Object Details**. The following object details are displayed:
   - Name
   - Storage Tier
   - Size
   - Content Type
   - ETag (entity tag)
   - Last Modified
7. Optionally, click **Download** to download the selected object.

To rename an object

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
2. Choose the compartment that contains the bucket that contains your object.
   A list of buckets is displayed.
3. Click the bucket name that contains your object.
4. Click **Objects** under **Resources**.
   A list of objects in the bucket is displayed.
5. For the object you want to rename, click the Actions icon (three dots), and then click **Rename**.
6. In the **Rename Object** dialog box, provide the new name for the object and an optional delimited directory structure prefix. For example, `p_94.jpg` or `/marathon/participants/p_94.jpg`.
   Avoid entering confidential information in the object name.



Warning

Buckets cannot store two objects that use identical names (case-sensitive). If you choose to rename an object using the name of another object in the same bucket, the object that originally used the name is overwritten.

7. Click **Save Changes**.

To restore objects from Archive Storage

Depending on the size of the object, it can take four or more hours to restore an object from Archive Storage. You cannot download an item until the item is fully restored.



Tip

You need OBJECT_RESTORE permissions to restore Archive Storage objects.

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
2. Choose the compartment your bucket is in.
   A list of buckets is displayed.
3. Click the bucket name that contains your object.
4. Click **Objects** under **Resources**.
   A list of objects in the bucket is displayed.
5. To restore a single object, click the Actions icon (three dots) to the right of the object you want to restore, and then click **Restore**. To restore multiple objects, select the check boxes to the left of each object you want to restore, then click **Restore**.

6. Optionally, specify the **Time Available for Download in Hours**.
   By default, you have 24 hours to download an object after restoration. However you can alternatively specify a download time of from 1 to 240 hours. You can find out how much download time is remaining by looking at **Available for Download** in object **Details** or by looking at the Actions icon (three dots) menu to the right of **Download**. Refresh the browser to obtain up-to-date remaining download time information.
   After the allotted download time expires, the object returns to Archive Storage.
7. Click **Restore Objects**.
   Error messages are generated if there is a problem with restoring the selected objects. You can optionally click **Retry failed restore option.**

To check the status of an Archive Storage object restoration

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
2. Choose the compartment your bucket is in.
   A list of buckets is displayed.
3. Click the bucket name that contains your object.
4. Click **Objects** under **Resources**.
   A list of objects in the bucket is displayed.
5. Click the Actions icon (three dots) to the right of the object you want to check the restoration or download status of, then click **Details**.
6. Check the **Status**.
   Status displays one of the following:
   - Archived
   - Restoring
   - Restored

To delete objects from a bucket

You can permanently delete an object from a bucket. There is no way to recover a deleted

object.

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
2. Choose the compartment that contains the bucket that contains the object or objects you want to delete.
   A list of buckets is displayed.
3. Click the bucket name that contains your object.
4. Click **Objects** under **Resources**.
   A list of objects in the bucket is displayed.
5. To delete a single object, click the Actions icon (three dots) to the right of the object you want to delete, and then click **Delete**. To delete multiple objects, select the check boxes to the left of each object you want to delete, and then click **Delete**.
6. Confirm when prompted.

Using the Command Line Interface (CLI)

For information about using the CLI, see Command Line Interface (CLI). For a complete list of flags and options available for CLI commands, see CLI Help.

To list objects in a bucket

Open a command prompt and run `oci os object list` to get a list of the objects in a bucket:

```
oci os object list -ns <object_storage_namespace> -bn <bucket_name>
```

By default, the following details are displayed for each object:

- Name
- Object size
- "Last Modified" timestamp
- MD5 hash

To get object details

Open a command prompt and run `oci os object head` to get object details:

```
oci os object head -ns <object_storage_namespace> -bn <bucket_name> --name <object_name>
```

The system output includes the following object details:

- ETag (entity tag)
- Content length (object body size)
- Custom metadata key-value pairs
- Storage tier
- MD5 hash
- Archival state

To upload an object to a bucket

Open a command prompt and run `oci os object put` to upload an object:

```
oci os object put -ns <object_storage_namespace> -bn <bucket_name> --file <file_location> --name
<object_name>  --no-multipart
```

Where *<file_location>* refers to a directory path like "C:\workspace\myfile.txt". If you want to use the filename as the uploaded object's name, you can omit the `--name` option. The resulting object name does not include the path information (for example, "C:\workspace\").

To add custom metadata key-value pairs, use the `--metadata` option:

```
oci os object put -ns <object_storage_namespace> -bn <bucket_name> --file <file_location> --name
<object_name>  --metadata <json_formatted_key-value_pairs> --no-multipart
```



Tip

The `--metadata` option requires that you provide complex type key-value pair input in valid JSON. See Passing Complex Input and Using a JSON File for Complex Input for more information about JSON formatting.

An object can be uploaded as a single part or as multiple parts. Here we describe a single part upload. For information on multipart uploads, see Using Multipart Uploads.

To bulk upload objects to a bucket

Open a command prompt and run `oci os object bulk-upload` to upload all files in a given directory (including files in subdirectories):

```
oci os object bulk-upload -ns <object_storage_namespace> -bn <bucket_name> --src-dir <source_
directory_location> --no-multipart
```

Where *<source_directory_location>* refers to a directory path like "C:\workspace\files_to_ upload\". If your source directory has subdirectories, the subdirectory names are prepended to the names of the files stored in those subdirectories, delimited with a forward slash (/) character. For example, if a file named "maple.jpg" is stored in the subdirectory "trees", when the file is uploaded, Object Storage assigns the name "trees/maple.jpg" to the resulting object.

To append a prefix string to the object names created by your uploads, use the `--object-prefix` option:

```
oci os object bulk-upload -ns <object_storage_namespace> -bn <bucket_name> --src-dir <source_
directory_location> --object-prefix <object_name_prefix_string> --no-multipart
```

For example:

```
oci os object bulk-upload -ns ansh8tvru7zp -bn apparel --src-dir C:\workspace\new_
items\bicycling\gloves\ --object-prefix /bicycling/gloves/ --no-multipart
```

To add custom metadata key-value pairs, use the `--metadata` option:

```
oci os object bulk-upload -ns <object_storage_namespace> -bn <bucket_name> --src-dir <source_
directory_location> --metadata <json_formatted_key-value_pairs> --no-multipart
```



Tip

The `--metadata` option requires that you provide complex type key-value pair input in valid JSON format. See Passing Complex Input and Using a JSON File for Complex Input for more information about JSON formatting.

To download an object from a bucket

Open a command prompt and run `oci os object get` to download an object:

```
oci os object get -ns <object_storage_namespace> -bn <bucket_name> --name <object_name> --file <file_
location>
```

Where *<file_location>* refers to a directory path like "C:\workspace\myfile.txt ".

To download an object using multipart download

Multipart object downloading is available using the byte-range request standard defined in RFC 7233, section 2.1.

Open a command prompt and run `oci os object get` with the `--range` option and the `bytes=<byte_range>` byte-range specifier to initiate a multipart download:

```
oci os object get -ns <object_storage_namespace> -bn <bucket_name> --name <object_name> --file <file_
location> --range bytes=<byte_range>
```

For example:

```
oci os object get -ns ansh81vru1zp -bn my_bucket --name my_object.mp4 --file /Users/me/my_object.mp4 --
range bytes=0-499
```

To bulk download all objects within a bucket

Open a command prompt and run `oci os object bulk-download` to download all the objects in a bucket:

```
oci os object bulk-download -ns <object_storage_namespace> -bn <bucket_name> --download-dir <download_directory_location>
```

Where *<download_directory_location>* refers to a directory path like "C:\workspace\objects\" where downloaded objects are saved. Object Storage creates this directory if it does not exist.

For a complete list of object bulk download options, see CLI Help.

To bulk download objects by object name prefix string

If you have named your objects with prefix strings, you can bulk download those objects in a bucket that match a specified prefix string. Open a command prompt and run `oci os object bulk-download` command with the `--prefix` option:

```
oci os object bulk-download -ns <object_storage_namespace> -bn <bucket_name> --download-dir <download_directory_location> --prefix <prefix_string>
```

Where *<download_directory_location>* refers to a directory path like "C:\workspace\objects\" where downloaded objects are saved. Object Storage creates this directory if it does not exist.

For example:

```
oci os object bulk-download -ns ansh8tvru7zp -bn apparel --download-dir C:\objects\ --prefix gloves_27
```

In the example above, an object named "gloves_27_A.jpg" is downloaded, while an object named "gloves_31_A.jpg" is not downloaded.

If you have named your objects so that they exist in Object Storage as a hierarchy, you can download objects at a given level and all sublevels by specifying the prefix that matches the level of your choosing:

```
oci os object bulk-download -ns <object_storage_namespace> -bn <bucket_name> --download-dir <download_directory_location> --prefix <level_1/level_2/>
```

The preceding command downloads the following files:

- *<level_1/level_2/object_name>*
- *<level_1/level_2/level_3/object_name>*
- *<level_1/level_2/level_3/level_4/object_name>*

To download only those objects at a given hierarchy level (and not objects in levels above or below), see To bulk download objects at a specified hierarchy level.

To bulk download objects at a specified hierarchy level

If you have named your objects so that they exist as a hierarchy, you can bulk download all the objects in a bucket at a specified hierarchy level (and not objects in levels above or below).

Open a command prompt and run `oci os object bulk-download` command with the `--prefix` and `--delimiter` flags:

```
oci os object bulk-download -ns <object_storage_namespace> -bn <bucket_name> --download-dir <download_directory_location> --prefix <level_1/level_2/> --delimiter /
```

Where *<download_directory_location>* refers to a directory path like "C:\workspace\objects\" where files downloaded objects are saved. If the directory you specify does not exist, Object Storage creates this directory.



Note

Currently, only the forward slash (/) is the supported delimiter for the `--delimiter` option.

The preceding command downloads all objects at *<level_2>* of the hierarchy. For example, the following object is downloaded:

<level_1/level_2/object_name>

The preceding command does not download objects in levels above or below *<level_2>*. For example, the following objects are not downloaded by the preceding command:

- *<level_1/object_name>*
- *<level_1/level_2/level_3/object_name>*
- *<level_1/level_2/level_3/level_4/object_name>*

To download objects at a given hierarchy level along with all objects in the associated hierarchy sublevels, see To bulk download objects by object name prefix string.

To rename an object

Open a command prompt and run `oci os object rename` to rename an object:

```
oci os object rename -ns <object_storage_namespace> -bn <bucket_name> --name <object_original_name> --
new-name <object_new_name>
```



Warning

Avoid entering confidential information in object name.

For example:

```
oci os object rename -ns ansh8tvru7zp -bn photo_collection --name /marathon/participants/p_93.jpg --new-
name /marathon/participants/p_94.jpg
```

To make the rename operation dependent on the object having a specific entity tag, use the `--src-obj-if-match-e-tag` option:

```
oci os object rename -ns <object_storage_namespace> -bn <bucket_name>--name <object_original_name> --
new-name <object_new_name> --src-obj-if-match-e-tag <etag_required_for_object_rename>
```

For example:

```
oci os object rename -ns ansh81vru1zp -bn my_bucket --name my_object.jpg --new-name my_renamed_
object.jpg --src-obj-if-match-e-tag 6672BECB67CCFFBCE0530292F20ZBACE
```

For rename operations where you intend to overwrite one object in a bucket with another, you can make the renaming dependent on having a specific entity tag. To do so, use the `--new-obj-if-match-e-tag` option:

```
oci os object rename -ns <object_storage_namespace> -bn <bucket_name> --name <source_object_name> --
new-name <name_of_object_to_be_overwritten> --new-obj-if-match-e-tag <etag_of_object_to_be_
overwritten>
```

For example:

```
oci os object rename -ns ansh8lvru1zp -bn my_bucket --name my_object.jpg --new-name my_renamed_
object.jpg --new-obj-if-match-e-tag 6672BECB67CCFFBCE0530292F20ZBACE
```

When renaming an object, you can prevent the system from overwriting another object in the same bucket by using the `--new-obj-if-none-match-e-tag *` option. This option prevents the renaming operation from completing if an object already exists with the `--new-name` value specified and the same entity tag of the source object.

```
oci os object rename -ns <object_storage_namespace> -bn <bucket_name> --name <source_object_name> --
new-name <new_name_for_object> --new-obj-if-none-match-e-tag *
```

For example:

```
 oci os object rename -ns ansh8lvru1zp -bn my_bucket --name my_object.jpg --new-name my_renamed_
object.jpg --new-obj-if-none-match-e-tag *
```

To restore an Archive Storage tier object



Tip

You need OBJECT_RESTORE permissions to restore Archive Storage objects.

Open a command prompt and run `oci os object restore` to restore an object from Archive Storage:

```
oci os object restore -ns <object_storage_namespace> -bn <archive_bucket_name> --name <archived_
object_name> [--hours <#_of_hours>]
```

By default, you have 24 hours to download an object after restoration. However, you can optionally specify `--hours` with an integer value of download time of from 1 to 240 hours.

To check the status of an Archive Storage object restoration

Open a command prompt and run `oci os object restore-status` to check the status of restoring an object from Archive Storage:

```
oci os object restore-status -ns <object_storage_namespace> -bn <archive_bucket_name> --name
<archived_object_name>
```

To delete an object

You can permanently delete an object. Open a command prompt and run `oci os object delete` to delete an object:

```
oci os object delete -ns <object_storage_namespace> -bn <bucket_name> --name <object_name>
```

To bulk delete all objects within a bucket

Open a command prompt and run `oci os object bulk-delete` to delete all the objects in a bucket:

```
oci os object bulk-delete -ns <object_storage_namespace> -bn <bucket_name>
```



Tip

To see a list of the files that would be deleted by a bulk delete command without actually deleting the files, use the `--dry-run` option.

To bulk delete objects by object name prefix string

If you named your objects with prefix strings, you can bulk delete those objects in a given

bucket by providing a prefix to match. Open a command prompt and run `oci os object bulk-delete` command with the `--prefix` option:

```
oci os object bulk-delete -ns <object_storage_namespace> -bn <bucket_name> --prefix <prefix_string>
```

For example:

```
oci os object bulk-delete -ns ansh8tvru7zp -bn apparel --prefix gloves_27
```

The preceding command deletes an object named `gloves_27_A.jpg`, but does not delete an object named `gloves_31_A.jpg`.

If you named your objects so that they exist as a hierarchy, you can bulk delete objects at a given level and all sublevels by specifying the prefix that matches the level of your choosing:

```
oci os object bulk-delete -ns <object_storage_namespace>-bn <bucket_name> --prefix <level_1/level_2/>
```

The preceding command deletes the following files:

- *<level_1/level_2/object_name>*
- *<level_1/level_2/level_3/object_name>*
- *<level_1/level_2/level_3/level_4/object_name>*

To delete only those objects at a given hierarchy level (and not objects in levels above or below), see To bulk delete objects at a specified hierarchy level.



Tip

To see a list of the files that will be deleted by a bulk delete command without actually deleting the files, use the `--dry-run` option.

To bulk delete objects at a specified hierarchy level

If you named your objects so that they exist in Object Storage as a hierarchy, you can bulk delete only those objects at a given hierarchy level (and not objects in levels above or below).

Open a command prompt and run the `oci os object bulk-delete` command with the `--prefix` and `--delimiter` flags:

```
oci os object bulk-delete -ns <object_storage_namespace> -bn <bucket_name> --prefix <level_1/level_2/>
--delimiter /
```



Note

Currently, only the forward slash (/) is the supported delimiter for the `--delimiter` option.

The preceding bulk delete command deletes the following object:

<level_1/level_2/>object_name

The preceding command does not bulk delete objects in levels above or below *<level_2>*. For example, the command would not delete the following objects:

- *<level_2/object_name>*
- *<level_1/level_2/level_3/object_name>*
- *<level_1/level_2/level_3/level_4/object_name>*

To delete objects at a given hierarchy level along with all objects in the associated hierarchy sublevels, see To bulk delete objects by object name prefix string.



Tip

To see a list of the files that will be deleted by a bulk delete command without actually deleting the files, use the `--dry-run` option.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Object Storage prepends the Object Storage namespace string and bucket name to the object name when constructing a URL for use with the API. Everything :

```
/n/<object_storage_namespace>/b/<bucket>/o/<object_name>
```

The object name is everything after the `/o/`, which could include hierarchy levels and prefix strings.

Use the following operations to manage objects:

- DeleteObject
- GetObject
- HeadObject
- ListObjects
- PutObject (see Special Instructions for Object Storage PUT for signing request requirements)
- RenameObject
- RestoreObjects

Copying Objects

This topic describes how to copy objects in Object Storage. You can copy objects to other buckets in the same region and to buckets in other regions.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have

permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

If you're new to policies, see Getting Started with Policies and Common Policies.



Warning

Object copy does not work if you do not authorize the Object Storage service to copy objects on your behalf. See Service Permissions for more information.

User Permissions

You must have the required access to both the source and destination buckets when performing an object copy. You must also have permissions to manage objects in the source and destination buckets.

For administrators:

- You can create a policy that lets the specified IAM group manage Object Storage namespaces, buckets, and their associated objects in all compartments in the tenancy:

  ```
  Allow group <IAM_group_name> to manage object-family in tenancy
  ```

- Alternatively, you can create policies that reduce the scope of access. For example, to let the specified group manage only buckets and objects in a particular compartment in the tenancy:

  ```
  Allow group <IAM_group_name> to manage buckets in compartment <compartment_name>
  ```

For more information about other alternatives for writing policies, see Details for Object Storage, Archive Storage, and Data Transfer.

Service Permissions

Because Object Storage is a regional service, you must authorize the Object Storage service for each region that will be carrying out copy operations on your behalf. For example, you

might authorize the Object Storage service in region us-ashburn-1 to manage objects on your behalf. Once you do this, you will be able to initiate the copy of an object stored in a us-ashburn-1 bucket to a bucket in any region, assuming that your user account has the required permissions to manage objects within the source and destination buckets.

To determine the region name value of an Oracle Cloud Infrastructure region, see Regions and Availability Domains .

For administrators:

To enable object copy, you must authorize the service to manage objects on your behalf:

- You can create a policy that authorizes the service in the specified region to manage Object Storage namespaces, buckets, and their associated objects in all compartments in the tenancy:

```
Allow service objectstorage-<region_name> to manage object-family in tenancy
```

- If you want to grant individual permissions to the service rather than use the policy verb `manage`, you can create a policy that further reduces the scope of access using one of the following statements:

```
Allow service objectstorage-<region_name> to {OBJECT_READ, OBJECT_INSPECT, OBJECT_CREATE,
OBJECT_OVERWRITE, OBJECT_DELETE} in tenancy
```

```
Allow service objectstorage-<region_name> to {OBJECT_READ, OBJECT_INSPECT, OBJECT_CREATE,
OBJECT_OVERWRITE, OBJECT_DELETE} in compartment <compartment_name>
```

Copy Object Work Requests

The Object Storage service handles copy requests asynchronously. The service creates a queue for copy requests, and then processes the requests as system resources become available. To provide visibility for in-progress copy operations, Object Storage creates a work request. You can track the progress of the copy operation by monitoring the status of the work request.

The work request statuses are:

ACCEPTED

The copy request is in the work request queue to be processed.

IN PROGRESS

The object is currently being copied.

SUCCEEDED

The copy operation has successfully completed.

CANCELING

The copy request is in the process of being canceled.

CANCELED

The copy request has been canceled.

FAILED

The copy operation has failed. Work requests that do not complete due to overwrite rules or insufficient user authorizations are assigned the failed status.

Copy Object Overwrite Rules

You can use overwrite rules to control the copying of objects based on their entity tag (ETag) values.

- **Overwrite destination object**: Use this option when you do not wish to limit a copy operation by an ETag value. This option is the default. This option can be used for any copy operation, regardless of whether or not it involves overwriting an existing object.
- **Do not overwrite any destination object**: Use this option to prevent the overwriting an existing copy of an object in the destination location, regardless of the destination object's ETag value.

- **Overwrite destination object only if it matches the specified ETag**: Use this option to prevent the accidental overwriting of an object in the destination location that does not have the specified ETag. When you use this option, the copy operation only succeeds if the ETag you supply when initiating the copy request matches the ETag of the destination object.
- **Copy object only if the source matches the specified ETag**: Use this option if you want the copy operation successful only if the ETag you supply when initiating the copy request matches the ETag of the source object. For objects that are intentionally updated and overwritten as part of data management activity, this option ensures that only the specified *version* of the object (as indicated by the ETag) is allowed to be copied. If the object's ETag value changes after the copy work request is created, but before the copy operation is executed, the copy operation will not complete.



Warning

If you overwrite an object, the operation cannot be undone.

Scope and Constraints

- Objects cannot be copied directly from Archive Storage. To copy objects that are currently in Archive Storage, you must first restore the object to the standard Object Storage tier. Objects can be copied directly to Archive Storage tier buckets from the standard Object Storage tier. When you copy an objects into an Archive Storage bucket, the copy of the object is immediately archived.
- You must specify an existing target bucket for the copy request. Buckets cannot be created automatically by the copy operation.
- When an object is copied, the destination object receives a new ETag value.
- If you rename, overwrite, or delete a source object while a copy operation is taking place, the copy operation fails and the destination object is not created or overwritten.

- For copy operations that overwrite an existing destination object, if you rename, overwrite, or delete the source object independently of the ongoing copy operation, the copy operation fails.
- Bulk copying is not supported. You must identify a single object in the request to be copied.

Using the Console

The Oracle Cloud Infrastructure Console consumes the REST API and is subject to the same considerations as any OCI client.

To make a copy of an object

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
2. Choose the compartment that contains the bucket that contains your object.
3. Select the bucket containing the object that you want to copy.
4. Click **Objects** under **Resources** to display a list of objects in the bucket.
5. For the object you want to copy (the source object), click the Actions icon (three dots), and then click **Copy**.
6. In the **Copy** dialog, enter the following:
    - **Destination Namespace**: The Object Storage Namespace of the destination bucket for your copied object. The namespace string of your tenancy is supplied as the default value.
    - **Destination Region**: The Oracle Cloud Infrastructure region containing the destination bucket for your copied object. Your tenancy must be subscribed to a region in order for you to copy an object to a bucket in that region.
    - **Destination Bucket**:The destination bucket for your copied object. This bucket must already exist. The copy operation cannot create a new bucket for the destination object.
    - **Overwrite Rule**: Select the overwrite rule appropriate for your copy request.

See Copy Object Overwrite Rules for information on the overwrite rule options.

- **Etag Match** (optional): Enter an ETag if you wish to use an overwrite rule that requires an ETag match on the source or destination object. This field is only enabled for copy requests that contain ETag matching overwrite rules.

7. Click **Copy Object**.
   A **Work request submitted** dialog confirms that your copy request was submitted successfully.

To monitor the status of an object copy work request

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
2. Choose the compartment containing your bucket.
3. Click the bucket name of the bucket containing the object being copied.
4. Click **Work Requests** under **Resources**.
   A list of work requests is displayed. The status of each work request is displayed, along with request details including object name, request ID, and the destination bucket's name, region, and namespace.

Using the Command Line Interface (CLI)

For information about using the CLI, see Command Line Interface (CLI). For a complete list of flags and options available for CLI commands, see CLI Help.

To make a copy of an object

Open a command prompt and run `oci os object copy` to copy an object:

```
oci os object copy --namespace-name <object_storage_namespace> --bucket-name <source_bucket_name> --
source-object-name <source_object> --destination-namespace <destination_namespace_string> --
destination-region <destination_region> --destination-bucket <destination_bucket_name> --destination-
object-name <destination_object_name>
```

For example:

```
oci os object copy --namespace-name ansh81vru1zp  --bucket-name photos --source-object-name
hummingbird.jpg --destination-namespace ansh81vru1zp --destination-region uk-london-1 --destination-
bucket UK_photos --destination-object-name hummingbird.jpg
```

For a complete list of object copy options, see CLI Help.

To get the status of an object copy work request

Use the `work-request get` command to get the status of an object copy work request using the work request ID. If you do not have the work request ID, you can get a list of work requests, including the request IDs, for a specified compartment using the `work-request list` command.

Open a command prompt and run `oci os work-request get` to get the status of a work request:

```
oci os work-request get --work-request-id <request_id>
```

For a complete list of work request options, see CLI Help.

To get a list of work requests for a compartment

Open a command prompt and run `oci os work-request list` to get a list of a work requests for a specified compartment:

```
oci os work-request list --compartment-id <compartment_id>
```

For a complete list of work request options, see CLI Help.

To cancel a copy object work request

Open a command prompt and run `oci os work-request cancel` to cancel a work request:

```
oci os work-request cancel --work-request-id <request_id>
```

For a complete list of work request options, see CLI Help.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these operations to view and manage work requests for copy object operations:

- CopyObject
- ListWorkRequests
- GetWorkRequest
- CancelWorkRequest

Using Pre-Authenticated Requests

Pre-authenticated requests provide a way to let users access a bucket or an object without having their own credentials, as long as the request creator has permissions to access those objects. For example, you can create a request that lets an operations support user upload backups to a bucket without owning API keys. Or, you can create a request that lets a business partner update shared data in a bucket without owning API keys.

When you create a pre-authenticated request, a unique URL is generated. Users in your organization, partners, or third parties can use this URL to access the Object Storage resource target identified in the pre-authenticated request.



Important

Carefully assess the business requirement for and the security ramifications of pre-authenticated access to a bucket or objects.

A pre-authenticated request URL gives anyone who has the URL access to the targets identified in the request for as long as the request is active. In addition to considering the operational needs of pre-authenticated access, it is equally important to manage its distribution.

Required Permissions

You need `PAR_MANAGE` permission access to the target bucket or object to create or manage pre-authenticated requests.

You also need permission to perform the action the pre-authenticated request is permitting. For example, if you are creating a pre-authenticated request for uploading an object, you must have both the `PAR_MANAGE` and the `OBJECT_CREATE` permissions in the target compartment.



Important

If the user who creates a pre-authenticated request is deleted or loses the `OBJECT_CREATE` permission after they created the request, then the request no longer works.

Options

When creating a pre-authenticated request, you have the following options:

- You can configure the name of a specific bucket that a user has write access to and can upload one or more objects to.
- You can configure the name of a specific object that a user can read from, write to, or read from and write to.
- You can configure the expiration date for the request.

Scope and Constraints

Understand the following scope and constraints regarding pre-authenticated requests:

- Users can't list bucket contents.
- There is no hard limit on the number of pre-authenticated requests that you can create.
- You can't edit a pre-authenticated request. If you want to change user access options in response to changing requirements, you need to create a new pre-authenticated request.
- The target and actions for a pre-authenticated request are based on its creator's permissions. The request is not, however, bound to the creator's account login credentials. If the creator's login credentials change, a pre-authenticated request is not affected.
- If the user who created a pre-authenticated request is deleted, then the request no longer works.
- You cannot delete a bucket that has a pre-authenticated request associated with that bucket or with an object in that bucket.

Working with Pre-Authenticated Requests

You can create, delete, or list pre-authenticated requests using the Console, using the CLI, or by using an SDK to access the API.



Important

The unique URL provided by the system when you create a pre-authenticated request is the only way a user can access the bucket or object specified as the request target. Copy the URL to durable storage. The URL is displayed only at the time of creation and cannot be retrieved later.

Using the Console

To create a pre-authenticated request for a bucket

See Service Limits for a list of applicable limits and instructions for requesting a limit increase.

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
2. Choose the compartment where the bucket is.
3. Click the bucket name.
4. Click **Pre-Authenticated Requests** under **Resources** to display the list of pre-authenticated requests.
5. Click **Create Pre-Authenticated Request**.
6. Provide the following information:
    - **Name:** The system generates a default, pre-authenticated request name that reflects the current year, month, day, and time, for example par-bucket-

20190307-0915. If you change this default or any other pre-authenticated request name, use letters, numbers, dashes, underscores, and periods. Type a name for the request, for example: **bucketPAR**. Avoid entering confidential information.

- **Pre-Authenticated Request Target**: Select **Bucket**.
- **Expiration**: Accept the system-generated expiration date or use the date and time editor to use a different expiration date and time.

7. Click **Create Pre-Authenticated Request**.
   After a request is created, the **Pre-Authenticated Request Details** dialog box displays the URL used to access the bucket, for example https://objectstorage.server.company.com/p/_QLD5xGz6vi7s6CkWnsqdgPJLpLE3a3sCBiDyoGCn3Q/n/tenancy/b/user_bucket_01-11-2019/o/.
8. Click **Copy** to copy the URL for future reference.



Note

The unique URL provided by the system when you create a pre-authenticated request is the only way a user can access the bucket or object specified as the request target. Copy the URL to durable storage. The URL is displayed only at the time of creation and cannot be retrieved later.

9. Click **Close**.

To create a pre-authenticated request for an object

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
2. Choose the compartment where the bucket is.
3. Click the bucket name.

4. Click **Objects** under **Resources** to display the list of objects.
5. For the object you want to create a pre-authenticated request, click the Actions icon (three dots), and then click **Create Pre-Authenticated Request**.
6. Provide the following information:
    - **Name:** The system generates a default, pre-authenticated request name that reflects the current year, month, day, and time, for example par-object-root-datatable.data-20190307-0948. If you change this default or any other pre-authenticated request name, use letters, numbers, dashes, underscores, and periods. Type a name for the request, for example: **bucketPAR**. Avoid entering confidential information.
    - **Pre-Authenticated Request Target:** Select **Object**.
    - **Object Name**: The name of the object that you want authenticated by this rule.
    - **Access Type:** Select one of the following.
        - Permit read on the object
        - Permit writes to the object
        - Permit reads on and writes to the object
    - **Expiration**: Accept the system-generated expiration date or use the date and time editor to a different expiration date and time.
7. a. Click **Create Pre-Authenticated Request**.
   After a request is created, the **Pre-Authenticated Request Details** dialog displays the URL used to access the object.
8. Click **Copy** to copy the URL for future reference.



Note

The unique URL provided by the system when you create a pre-authenticated request is the only way a user can access the bucket or object specified as the request target. Copy the URL to durable storage. The URL is displayed only at the time of creation and cannot be retrieved later.

9. Click **Close**.

To copy a pre-authenticated request ID

To copy the ID for a pre-authenticated request to the clipboard, do the following:

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
2. Choose the compartment where the bucket is.
3. Click the bucket name.
4. Click **Pre-Authenticated Requests** under **Resources** to display the list of pre-authenticated requests.
5. For the pre-authenticated request for which you want to copy its ID, click the Actions icon (three dots), and then click **Copy Pre-Authenticated Request ID**.
   The ID for the selected pre-authentication request is copied to the clipboard, for example rNQxBFULWOSKgs+6tgddhj9GscuD0h4Q2exvW+I5l9I=:root-data-table.data.

Using the Command Line Interface (CLI)

For information about using the CLI, see Command Line Interface (CLI). For a complete list of flags and options available for CLI commands, see CLI Help.

To create a pre-authenticated request for a bucket

```
oci os preauth-request create -ns <object_storage_namespace> -bn <bucket_name> --name
<preauthenticated_request_name> --access-type AnyObjectWrite --time-expires <timestamp>
```

Note the following:

- To create a pre-authenticated request for a bucket, use the `AnyObjectWrite` enum value with the `--access-type` flag. Pre-authenticated requests for buckets permit writes to the bucket by default.
- The *<timestamp>* value must be an RFC 3339 timestamp. For example: 2017-09-01T00:09:51.000+02:00.



Note

The unique URL provided by the system when you create a pre-authenticated request is the only way a user can access the bucket or object specified as the request target. Copy the URL to durable storage. The URL is displayed only at the time of creation and cannot be retrieved later.

To create a pre-authenticated request for an object

```
oci os preauth-request create -ns <object_storage_namespace> -bn <bucket_name> --name
<preauthenticated_request_name> --access-type <enum_value> --time-expires <timestamp> -on <object_
name_or_null>
```

Where *<enum_value>* is one of the following:

- ObjectRead
- ObjectWrite
- ObjectReadWrite
- AnyObjectWrite

The *<timestamp>* value must be an RFC 3339 time stamp. For example: 2017-09-01T00:09:51.000+02:00.

Avoid entering confidential information in the pre-authenticated request name.



Note

The unique URL provided by the system when you create a pre-authenticated request is the only way a user can access the bucket or object specified as the request target. Copy the URL to durable storage. The URL is displayed only at the time of creation and cannot be retrieved later.

To list a pre-authenticated request

```
oci os preauth-request list -ns <object_storage_namespace> -bn <bucket_name>
```

To get a pre-authenticated request

```
oci os preauth-request get -ns <object_storage_namespace> -bn <bucket_name> --par-id
<preauthenticated_request_id>
```

To delete a pre-authenticated request

```
oci os preauth-request delete -ns <object_storage_namespace> -bn <bucket_name> --par-id
<preauthenticated_request_id>
```

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operations to work with pre-authenticated requests:

- CreatePreauthenticatedRequest
- DeletePreauthenticatedRequest
- GetPreauthenticatedRequest
- ListPreauthenticatedRequests

Using Multipart Uploads

The Oracle Cloud Infrastructure Object Storage service supports multipart uploads for more efficient and resilient uploads, especially for large objects. You can perform multipart uploads using the API, the Software Development Kits and Command Line Interface, or the Command Line Interface (CLI). With multipart uploads, individual parts of an object can be uploaded in parallel to reduce the amount of time you spend uploading. Multipart uploads performed through the API can also minimize the impact of network failures by letting you retry a failed part upload instead of requiring you to retry an entire object upload.

Multipart uploads can accommodate objects that are too large for a single upload operation. Oracle recommends that you perform a multipart upload to upload objects larger than 100 MiB. The maximum size for an uploaded object is 10 TiB. Object parts must be no larger than 50 GiB. For very large uploads performed through the API, you have the flexibility of pausing between the uploads of individual parts, and resuming the upload as your schedule and resources allow.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let Object Storage admins manage buckets and objects lets the specified group do everything with buckets and objects.

If you are new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for Object Storage, see Details for Object Storage, Archive Storage, and Data Transfer.

Monitoring Resources

You can monitor the health, capacity, and performance of your Oracle Cloud Infrastructure resources by using metrics, alarms, and notifications. For more information, see Monitoring Overview and Notifications Overview.

For more information about monitoring multipart uploads, see Object Storage Metrics.

Using the Multipart Upload API

A multipart upload performed using the API consists of the following steps:

1. Initiating an upload
2. Uploading object parts
3. Committing the upload

Before you use the multipart upload API, you are responsible for creating the parts to upload. Object Storage Provides API operations for the remaining steps. The service also provides API operations for listing in-progress multipart uploads, listing the object parts in an in-progress multipart upload, and aborting in-progress multipart uploads initiated through the API.

Here we describe the API steps at a high level, but you can refer to the API Reference for specifics about supported API calls.

Creating Object Parts

With multipart upload, you split the object you want to upload into individual parts. Individual parts can be as large as 50 GiB or as small as 10 MB. (Object Storage waives the minimum part size restriction for the last uploaded part.) Decide what part number you want to use for each part. Part numbers can range from 1 to 10,000. You do not need to assign contiguous

numbers, but Object Storage constructs the object by ordering part numbers in ascending order.

Initiating an Upload

After you finish creating object parts, initiate a multipart upload by making a CreateMultipartUpload REST API call. Provide the object name and any object metadata. Object Storage Responds with a unique upload ID that you must include in any requests related to this multipart upload. Object Storage also marks the upload as active. The upload remains active until you explicitly commit it or abort it.

Uploading Object Parts

Make an UploadPart request for each object part upload. In the request parameters, provide the Object Storage namespace, bucket name, upload ID, and part number. In the request body, include the object part. Object parts can be uploaded in parallel and in any order. When you commit the upload, Object Storage uses the part numbers to sequence object parts. Part numbers do not have to be contiguous. If multiple object parts are uploaded using the same upload ID and part number, the last upload overwrites the part and is committed when you call the CommitMultipartUpload API.

Object Storage returns an ETag (entity tag) value for each part uploaded. You need both the part number and corresponding ETag value for each part when you commit the upload.

If you have network issues, you can restart a failed upload for an individual part. You do not need to restart the entire upload. If, for some reason, you cannot perform an upload all at once, multipart upload lets you continue uploading parts at your own pace. While a multipart upload is still active, you can keep adding parts as long as the total number is less than 10,000.

You can check on an active multipart upload by listing all parts that have been uploaded. (You cannot list information for an individual object part in an active multipart upload.) The ListMultipartUploadParts operation requires the Object Storage namespace, bucket name, and upload ID. Object Storage responds with information about the parts associated with the specified upload ID. Parts information includes the part number, ETag value, MD5 hash, and part size (in bytes).

Similarly, if you have multiple multipart uploads occurring simultaneously, you can see what uploads are in-progress. Make an ListMultipartUploads API call to list active multipart uploads in the specified Object Storage namespace and bucket.

Charges for parts storage begin accruing when you upload data.

Committing the Upload

When you have uploaded all object parts, commit the upload. Use the CommitMultipartUpload request parameters to specify the Object Storage namespace, bucket name, and upload ID. Include the part number and corresponding ETag value for each part in the body of the request. When you commit the upload, Object Storage constructs the object from its constituent parts. The object is stored in the specified bucket and Object Storage namespace. You can treat it like you would any other object. Garbage collection releases storage space occupied by any part numbers you uploaded, but did not include in the CommitMultipartUpload request.

You cannot list or retrieve parts from a completed upload. You cannot append or remove parts from the completed upload either. If you want, you can replace the object by initiating a new upload.

If you decide to abort a multipart upload instead of committing it, wait for in-progress part uploads to complete and then use the AbortMultipartUpload operation. If you abort an upload while part uploads are still in progress anyway, Object Storage cleans up both completed and in-progress parts. Upload IDs from aborted multipart uploads cannot be reused.

API Documentation

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operations to manage multipart uploads:

- AbortMultipartUpload
- CommitMultipartUpload

- CreateMultipartUpload
- ListMultipartUploadParts
- ListMultipartUploads
- UploadPart (see Special Instructions for Object Storage PUT for signing request requirements)

Using the CLI

When you perform a multipart upload using the CLI, you do not need to split the object into parts as you are required to do by the API. Instead, you specify the part size of your choice, and Object Storage splits the object into parts and performs the upload of all parts automatically. You can choose to set the maximum number of parts that can be uploaded in parallel. By default, the CLI limits the number of parts that can be uploaded in parallel to three. When using the CLI, you do not have to perform a commit when the upload is complete.

You can also use the CLI to list in-progress multipart uploads, and to abort multipart uploads initiated through the API.

For information about using the CLI, see Command Line Interface (CLI). For a complete list of flags and options available for CLI commands, see CLI Help.

To perform a multipart upload using the CLI

To upload an object, open a command prompt and run `oci os object put` with the `--part-size` flag. The `--part-size` value represents the size of each part in mebibytes (MiB). Object Storage waives the minimum part size restriction for the last uploaded part. The `--part-size` value must be an integer.

Optionally, you can use the `--parallel-upload-count` flag to set the maximum number of parallel uploads allowed.

```
oci os object put -ns <object_storage_namespace> -bn <bucket_name> --file <file_location> --name
<object_name> --part-size <upload_part_size_in_MB> --parallel-upload-count <maximum_number_parallel_
uploads>
```

For more information on the `oci os object put` command, see To upload an object to a bucket.

To list in-progress multipart uploads for a bucket

Open a command prompt and run `oci os multipart list` to list all in-progress multipart uploads for a bucket:

```
oci os multipart list -ns <object_storage_namespace> -bn <bucket_name>
```



Tip

See CLI Help for command options to control the pagination of the list output.

To abort an uncommitted multipart upload initiated through the API

Open a command prompt and run `oci os multipart abort` to abort an in-progress multipart upload initiated through the API:

```
oci os multipart abort -ns <object_storage_namespace> -bn <bucket_name> --object-name <object_name> --upload-id <upload_ID>
```



Tip

The CLI interface asks you to confirm the deletion request. To abort without the confirmation prompt, use the `--force` flag.

Using Object Lifecycle Management

Object Lifecycle Management lets you automatically manage the archiving and deletion of objects. By using Object Lifecycle Management to manage your Object Storage and Archive Storage data, you can reduce your storage costs and the amount of time you spend managing data.

Object Lifecycle Management works by defining rules that instruct Object Storage to archive or delete objects on your behalf within a given bucket. A bucket's lifecycle rules are collectively known as an object lifecycle policy. For example, you could have Object Storage automatically move objects to Archive Storage 30 days after creation, and then automatically delete the archived objects 120 days after creation.

Each Object Storage or Archive Storage bucket can have a single lifecycle policy consisting of up to 1,000 rules. Rules can have object name prefix and pattern matching conditions. You can create, edit, delete, enable, and disable individual rules in the Console as needed. To update a lifecycle policy using the CLI or API, you need to overwrite the entire policy with a new policy that is inclusive of all the policy rules that you want to apply to the bucket.

Required IAM Policy



Important

You cannot use Object Lifecycle Management until you authorize the Object Storage service to archive and delete objects on your behalf. See Service Permissions for more information.

User Permissions

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have

permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

The policy Let Object Storage admins manage buckets and objects lets the specified group do everything with buckets and objects, including adding and managing lifecycle policies. See Details for Object Storage, Archive Storage, and Data Transfer for more information on Object Storage user permissions.

Service Permissions

To execute object lifecycle policies, you must authorize the service to archive and delete objects on your behalf. To do so, create the following policy in the root compartment of your tenancy:

```
Allow service objectstorage-<region_name> to manage object-family in compartment <compartment_name>
```

Because Object Storage is a regional service, you must authorize the Object Storage service in each region you use lifecycle policies. Object Storage ensures that your data is not read from any unauthorized region.

If you don't have permissions to write policies for the root compartment of your tenancy, contact your Oracle Cloud Infrastructure administrator. To determine the region name value of an Oracle Cloud Infrastructure region, see Regions and Availability Domains.

If you want to grant individual permissions to the service rather than use the policy verb `manage`, you can use the following syntax:

```
Allow service objectstorage-<region_name> to {BUCKET_INSPECT, BUCKET_READ, OBJECT_INSPECT, OBJECT_
CREATE,  OBJECT_DELETE} in compartment <compartment_name>
```

If you're new to policies, see Getting Started with Policies and Common Policies.

Options

When creating object lifecycle policy rules, you have the following options:

- When a lifecycle rule is created, the system generates a default name for that rule, for example **lifecycle-rule-20190321-1559**. This rule name identifies the current year,

month, day, and time that the rule was created. You can use that system-generated name for your new rule or you can specify a different name for it.

- You can use a rule to either archive or delete objects and specify the number of days until the specified action is taken.
- You can apply a rule to all objects in a bucket. Alternatively, you can use object name filters to specify which objects the lifecycle rule applies to. You can select objects using both object name prefixes and pattern matching. See Using Object Name Filters for details.
- You can decide whether a new rule is enabled or disabled upon creation.

Using Object Name Filters

Use object name filters to specify which objects the lifecycle rule applies to.



Important

If you want the rule to apply to all objects in the bucket, do not specify any object name filters.

You can add object filters in any order. Object Lifecycle Management evaluates the precedence of the rules as follows:

1. Pattern exclusions
2. Pattern inclusions
3. Prefix inclusions

Using Prefix Matching to Filter Objects

When naming objects, you can use prefix strings without a delimiter so that certain bulk operations can be performed by matching on the prefix portion of the object name. For example, in the object names below, the string `gloves_27_` serves as a prefix for matching purposes when performing lifecycle management archive or deletions:

```
gloves_27_dark_green.jpg
gloves_27_light_blue.jpg
gloves_27_deep_purple.jpg
gloves_27_bright_orange.jpg
```

See Object Naming Using Prefixes and Hierarchies for complete object naming details.

Using Pattern Matching to Filter Objects

Object Storage supports the following pattern matching characters to either include or exclude objects:

Character	Description	Pattern Examples	Matches	Doesn't Match
*	Matches 0 or more characters	*.tmp	foo.tmp foo/bar/baz.tmp	tmp Atmp
		*.xls	.xls /home/user/file.xlsx	xls .xl
		/archive/*	/archive/sub/dir/ /archive/1/2/3/4/foo.txt	/src/archive/a archive/b
?	Matches any one character	X?Z	XyZ X_Z	XZ XYYZ
\	Escapes the next character	\\dir\\sub*	\dir\sub\ABC \dir\sub\	dir\sub\abc dirsub

Character	Description	Pattern Examples	Matches	Doesn't Match
[...]	Matches a group of characters, which can be: • A set of characters, for example: [Zafg9@]. Matches any character in the brackets. • A range of characters, for example: [a-f]. Matches any character in the range. Note that: ◦ [a-f] is equivalent to [abcdef]. ◦ For character ranges only the CHARACTER-CHARACTER pattern	[-ab3]	- a b 3	-a -ab 3b

Character	Description	Pattern Examples	Matches	Doesn't Match
	is supported: ■ [ab-yz] is not valid ■ [a-mn-z] is not valid ◦ Character ranges cannot start with ^ or : ◦ To include a hyphen (-) in the range, make it the first or last character.			
		backup.tar.gz.[0-9]	backup.tar.gz.0 backup.tar.gz.5 backup.tar.gz.9	backup.tar.gz.10 backup.tar.gz
		page-[0-9]*	page-0 page-2 page-22 page-2X	page- page-A1
		\[a-z\]	[a-z]	a z [a-z

Patterns are limited to 1024 characters. The following are examples of invalid patterns:

- \
- [^a-z]
- [z-a]
- [:isalpha:]

Scope and Constraints

Understand the following scope and constraints regarding object lifecycle policies:

- When you create a lifecycle policy for a bucket, Object Storage applies that policy to all objects that exist in the bucket unless you add object name filters.
- A rule that deletes an object always takes priority over a rule that would archive that same object.
- When creating a lifecycle policy rule that deletes objects from Archive Storage, Archive Storage has a minimum retention requirement of 90 days. Objects deleted from Archive Storage that have not met the 90-day retention minimum are billed for 90 days of storage.
- You can create up to 1,000 lifecycle rules per bucket.

Working with Object Lifecycle Management Policies

You can create, delete, edit, or disable lifecycle policy rules using the Console, the Command Line Interface (CLI), an SDK, or the API.



Warning

Objects deleted on your behalf by lifecycle policies cannot be recovered. Be sure when creating and editing your lifecycle policies that you are not unintentionally deleting data you want to retain. Oracle recommends that you test your lifecycle policy on development data before using the policy in production.

Using the Console

To create a lifecycle policy rule

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
2. Choose the compartment containing bucket for which you want to create a lifecycle rule.
3. Click the bucket name.
4. Click **Lifecycle Policy Rules** under **Resources** to access the lifecycle policy rule list.
5. Click **Create Rule**.
6. Provide the following information:
   - **Name:** Required. The system generates a default rule name that reflects the current year, month, day, and time, for example **lifecycle-rule-20190321-1559**. If you change this default to any other rule name, use letters, numbers, dashes, underscores, and periods. Do not include any confidential information.
   - **Lifecycle Action:** Select rule type **Archive** or **Delete**.
   - **Number of Days:** The number of days until the specified action is taken.
7. Optionally, you can add one or more **Object Name Filters** to specify which objects the lifecycle rule applies to. You can choose objects using prefixes and pattern matching. If no object name filters are specified, the rule applies to all objects in the bucket.

To create an object name filter:

a. Click **Add Filter**.
b. Select the **Filter Type**.
c. Enter the **Filter Value**.
d. Click **Add Another Filter** to add as many filters as you need for this rule.

8. Select whether the rule is enabled or disabled upon creation using the **State** selector.
9. Click **Create**.

To edit a lifecycle policy rule

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
2. Choose the compartment where the bucket is.
3. Click the bucket name.
4. Click **Lifecycle Policy Rules** under **Resources** to access the rule list.
5. For the rule you want to edit, click the Actions icon (three dots), and then click **Edit**.
6. In the **Edit Lifecycle Rule** dialog box, edit the following as needed for each rule you want to change:
   - **Name:** A user-friendly name for the rule. Avoid entering confidential information.
   - **Lifecycle Action:** Rule type **Archive** or **Delete**.
   - **Number of Days:** The number of days until the specified action is taken.
   - **Object Name Filters:** Edit, delete, or add a prefix or pattern filter.
7. Click **Save Changes**.

To enable, disable, or delete a lifecycle policy rule

You can disable and enable a rule on demand using the Console. The system stops the execution of disabled or deleted rules immediately.

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
2. Choose the compartment where the bucket is.
3. Click the bucket name.
4. Click **Lifecycle Policy Rules** under **Resources** to access the rule list.
5. For the rule you want to manage, click the Actions icon (three dots), and then click one of the following:
   - **Enable** (only displays if the rule is disabled)
   - **Disable** (only displays if the rule is enabled)
   - **Delete**

Using the Command Line Interface (CLI)

For information about using the CLI, see Command Line Interface (CLI). For a complete list of flags and options available for CLI commands, see CLI Help.

To create or replace a lifecycle policy for a bucket

Open a command prompt and run `oci os object-lifecycle-policy put` to create or replace the object lifecycle policy for a bucket. To edit individual rules, replace the bucket's existing policy with a new version of the policy that includes the changes to your rules.

```
oci os object-lifecycle-policy put -ns <object_storage_namespace> -bn <bucket_name> --items <json_formatted_lifecycle_policy>
```



Tip

The `--items` option requires that you provide key-value pair input as valid formatted JSON. See Passing Complex Input and Using a JSON File for Complex Input for information on JSON formatting.

For example, the following lifecycle policy archives objects after 30 days and deletes them after 180 days:

```
oci os object-lifecycle-policy put -ns MyNamespace -bn MyBucket --items '[
    {
        "action": "ARCHIVE",
        "is-enabled": true,
        "name": "ArchiveAfter30Days",
        "object-name-filter": {
          "exclusion-patterns": [
            "*.jpg"
          ],
          "inclusion-patterns": [
            "*.doc"
          ],
          "inclusion-prefixes": [
            "documents/"
          ]
        },
        "time-amount": 30,
        "time-unit": "DAYS"
      },
      {
        "action": "DELETE",
        "is-enabled": true,
        "name": "DeleteAfter180Days",
        "object-name-filter": {
          "exclusion-patterns": null,
          "inclusion-patterns": null,
          "inclusion-prefixes": null
        },
        "time-amount": 180,
        "time-unit": "DAYS"
      }
]'
```

On Windows, to pass complex input to the CLI as a JSON string, you must enclose the entire block in double quotes. Inside the block, each double quote for the key and value strings must be escaped with a backslash (\) character.

For example:

```
oci os object-lifecycle-policy put -ns MyNamespace -bn MyBucket --items "
[{\"action\":\"ARCHIVE\",\"is-enabled\":true,\"name\":\"Archive After 30 Days\",\"object-name-
filter\":{\"exclusion-patterns\":[\"*.jpg\"],\"inclusion-patterns\":[\"*.doc\"],\"inclusion-
prefixes\":[\"documents/\"]},\"time-amount\":30,\"time-unit\":\"DAYS\"},{\"action\":\"DELETE\",\"is-
enabled\":true,\"name\":\"DeleteAfter180Days\",\"object-name-filter\":{\"exclusion-
patterns\":null,\"inclusion-patterns\":null,\"inclusion-prefixes\":null},\"time-amount\":180,\"time-
unit\":\"DAYS\"}]"
```

To delete a bucket's lifecycle policy

Open a command prompt and run `oci os object-lifecycle-policy delete` to delete a

bucket's object lifecycle policy.

```
oci os object-lifecycle-policy delete -ns <object_storage_namespace> -bn <bucket_name>
```

To get a bucket's lifecycle policy

Open a command prompt and run `oci os object-lifecycle-policy get` to get a bucket's object lifecycle policy.

```
oci os object-lifecycle-policy get -ns <object_storage_namespace> -bn <bucket_name>
```

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operations to manage object lifecycle policies:

- PutObjectLifecyclePolicy
- GetObjectLifecyclePolicy
- DeleteObjectLifecyclePolicy

Object Storage Metrics

You can monitor the health, capacity, and performance of your buckets and objects by using metrics, alarms, and notifications.

This topic describes the metrics emitted by the metric namespace `oci_objectstorage` (the Object Storage service).

Resources include buckets and objects.

Overview of the Object Storage Service Metrics

Object Storage can store an unlimited amount of unstructured data of any content type, including analytic data and rich content, like images and videos. The Object Storage service metrics help you measure the amount of storage you're using, the number of objects being stored, and the size of any objects included in incomplete parts of a multipart upload. For more information, see Overview of Object Storage and Using Multipart Uploads.

Prerequisites

IAM policies: To monitor resources, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. The policy must give you access to the monitoring services as well as the resources being monitored. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in. For more information on user authorizations for monitoring, see the Authentication and Authorization section for the related service: Monitoring or Notifications.

Available Metrics: oci_ objectstorage

The metrics listed in the following table are automatically available for any buckets you create. You do not need to enable monitoring on the resource to get these metrics. However, you must have an object stored in a bucket to get any metrics. Buckets with no objects elicit no metric data.

Each metric includes the following dimensions:

RESOURCEID

The OCID of the bucket to which the metric applies.

RESOURCEDISPLAYNAME

The name of the bucket.

TIER

The current storage tier of the object: standard or archived.

Metric	Metric Display Name	Unit	Description	Dimensions
`StoredBytes`	**Bucket Size**	bytes	The size of the bucket, excluding any multipart upload parts that have not been discarded (aborted) or committed.	`resourceID` `resourceDisplayName` `tier`
`ObjectCount`	**Number of Objects**	count	The count of objects in the bucket, excluding any multipart upload parts that have not been discarded (aborted) or committed.	
`UncommittedParts`	**Incomplete MultiPart Upload Size**	bytes	The size of any multipart upload parts that have not been discarded (aborted) or committed. **Tip**: You must create a custom query to see this metric. See To view UncommittedParts in a chart.	

Using the Console

To view default metric charts for a single bucket

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
2. Choose the **Compartment** that contains the bucket you want to view, and then click the bucket's name.
3. In the **Resources** menu, click **Metrics**.
   The **Metrics** page displays a default set of charts for the current bucket.

For more information about monitoring metrics and using alarms, see Monitoring Overview. For information about notifications for alarms, see Notifications Overview.

To view default metric charts by tier

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Service Metrics**.
2. For **Metric Namespace**, select **oci_objectstorage**.
3. For **Dimensions**, click **Add**.
4. For **Dimension Name** select **tier** and then select a **Dimension Value**.
5. Click **Done**.
   The **Service Metrics** page displays a default set of charts for the selected metric namespace. For more information about the emitted metrics, see the foregoing table. You can also use the Monitoring service to create custom queries.

For more information about monitoring metrics and using alarms, see Monitoring Overview. For information about notifications for alarms, see Notifications Overview.

To view UncommittedParts in a chart

You must create a custom query to see this metric.

1. Open the navigation menu. Under **Solutions, Platform and Edge**, go to **Monitoring** and click **Service Metrics**.
   The **Metrics Explorer** page displays an empty chart with fields to build a query.
2. Select a compartment and fill in the following fields.
   - **Metric Namespace**: **oci_objectstorage**
   - **Metric Name**: **UncommittedParts**
3. Refine your query.
   For instructions, see To create a query.
4. Click **Update Chart**.
   The chart shows the results of your new query. You can optionally add more queries by clicking **Add Query** below the chart.

For more information about monitoring metrics and using alarms, see Monitoring Overview. For information about notifications for alarms, see Notifications Overview.

Using the API

Use the following APIs for monitoring:

- Monitoring API for metrics and alarms
- Notifications API for notifications (used with alarms)

Hadoop Support

Using the HDFS connector, you can run Hadoop or Spark jobs against data in the Oracle Cloud Infrastructure Object Storage service. The connector has the following features:

- Supports read and write data stored in Object Storage
- Is compatible with existing buckets of data
- Is compatible with Hadoop 2.7.2

For information about downloading, configuring, and using the HDFS connector, see HDFS Connector for Object Storage.

Designating Compartments for the Amazon S3 Compatibility and Swift APIs

In the Oracle Cloud Infrastructure Object Storage service, a bucket is a container for storing objects in a compartment within an Object Storage namespace. A bucket is associated with a single compartment and data is stored as objects in buckets.

In addition to the native Object Storage APIs, Object Storage provides API support for both Amazon S3 Compatibility API and Swift API. However these APIs do not understand the Oracle Cloud Infrastructure concept of a compartment. By default, buckets created using the Amazon S3 Compatibility API or the Swift API are created in the root compartment of the Oracle Cloud Infrastructure tenancy. Instead, you can designate a different compartment for the Amazon S3 Compatibility API or Swift API to create buckets in.

When you designate a different compartment to use for the Amazon S3 Compatibility API or Swift API, any new buckets you create using the Amazon S3 Compatibility API or the Swift API are created in this newly designated compartment. Buckets previously created in a different compartment are not automatically moved to the newly designated compartment. See Managing Buckets if you want to move previously created buckets to this newly designated compartment.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

Compartments have policies that indicate what actions a user can perform on a bucket and all the objects in the bucket.

For administrators:

- To change the default compartments for Amazon S3 Compatibility API and Swift API, a user must belong to a group with `NAMESPACE_UPDATE` permissions.
- To see the current default compartments for Amazon S3 Compatibility API and Swift API, a user must belong to a group with `NAMESPACE_READ` permissions.
- To move a bucket to a different compartment, a user must belong to a group with `BUCKET_UPDATE` and `BUCKET_CREATE` permissions in the source compartment, and `BUCKET_CREATE` permissions in the target compartment.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for buckets and objects, see Details for Object Storage, Archive Storage, and Data Transfer.

Viewing and Specifying Designated Compartments

You can view the current default compartment designations for Amazon S3 Compatibility API and Swift API data. If your permissions allow, you can also change the Amazon S3 Compatibility API and Swift API compartment designations.

Designated compartment names:

- Must be unique across all the compartments in your tenancy.
- Can be from 1 to 100 characters in length.
- Must not contain confidential information.
- Valid are letters (upper or lower case), numbers, hyphens, and underscore.

Using the Console

To view your Amazon S3 Compatibility API and Swift API compartment designations

Open the **User** menu () and click **Tenancy:** *<your_tenancy_name>*.

Your default compartment designations for the APIs are listed under **Object Storage Settings**.

To edit your tenancy's Amazon S3 Compatibility API and Swift API compartment designations

1. Open the **User** menu () and click **Tenancy:** *<your_tenancy_name>*.
2. Click **Edit Object Storage Settings**.
3. In the **Edit Object Storage Settings** dialog:
   - Select the compartment that you want for the **Amazon S3 Compatibility API Designated Compartment** from the drop-down menu.
   - Select the compartment that you want for the **Swift API Designated Compartment** from the drop-down menu.
4. Click **Save**.
   The new **Object Storage Settings** are displayed.

Using the Command Line Interface (CLI)

For information about using the CLI, see Command Line Interface (CLI). For a complete list of flags and options available for CLI commands, see CLI Help.

To get your tenancy's Amazon S3 Compatibility API and Swift API compartment designations

Open a command prompt and run `oci os ns get-metadata` to get your compartment designations.

For example:

```
oci os ns get-metadata --namespace <object_storage_namespace>
```

To update your tenancy's Amazon S3 Compatibility API compartment designation

Open a command prompt and run `oci os ns update-metadata` to update your compartment designation for Amazon S3 Compatibility API.

For example:

```
oci os ns update-metadata --namespace <object_storage_namespace> --default-s3-compartment-id <your_oci_compartment_id>
```

Where *<your_oci_compartment_id>* specifies a compartment that is not the root compartment of your tenancy.

To update your tenancy's Swift API compartment designations

Open a command prompt and run `oci os ns update-metadata` to update your compartment designation for Swift API.

For example:

```
oci os ns update-metadata --namespace <object_storage_namespace> --default-swift-compartment-id <your_oci_compartment_id>
```

Where *<your_oci_compartment_id>* specifies a compartment that is not the root compartment of your tenancy.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operation to get your default Amazon S3 Compatibility API and Swift API compartment designations, and change those compartment designations:

- GetNamespaceMetadata
- UpdateNamespaceMetadata

Amazon S3 Compatibility API

Using the Amazon S3 Compatibility API, customers can continue to use their existing Amazon S3 tools (for example, SDK clients) and partners can make minimal changes to their applications to work with Object Storage. The Amazon S3 Compatibility API and Object Storage data sets are congruent. If data is written to the Object Storage using the Amazon S3 Compatibility API, the data can be read back using the native Object Storage API and vice versa.

Differences between the Object Storage API and the Amazon S3 Compatibility API

The Object Storage Service provided by Oracle Cloud Infrastructure and Amazon S3 use similar concepts and terminology. In both cases, data is stored as objects in buckets. The differences are in the implementation of features and tools for working with objects.

The following highlights the differences between the two storage technologies:

- **Compartments**
  Although Amazon S3 doesn't use compartments, any buckets created using the Amazon S3 Compatibility API are created in the root compartment of the Oracle Cloud Infrastructure tenancy.
- **Global bucket namespace**
  Object Storage doesn't use a global bucket namespace. Bucket names must be unique within the context of a namespace, but bucket names can be repeated across namespaces or across regions. Each tenant is associated with one default namespace that spans all compartments within a region.
- **Encryption**
  The Oracle Cloud Infrastructure Object Storage service encrypts all data at rest by default. Encryption can't be turned on or off using the API.
- **Object Level Access Control Lists (ACLs)**
  Oracle Cloud Infrastructure does not use ACLs for objects. Instead, IAM policies are used to manage access to compartments, buckets, and objects.

For more information, see Overview of the Object Storage Service.

Amazon S3 Compatibility API Support

Amazon S3 Compatibility API support is provided at the bucket level and object level.

Bucket APIs

The following bucket APIs are supported:

- DeleteBucket
- GetLocation
- HeadBucket
- ListObjects
- PutBucket

Object APIs

The following object APIs are supported:

- DeleteObject
- GetObject
- HeadObject
- PutObject

Tagging APIs

The following tagging APIs are supported:

- DeleteBucketTagging
- GetBucketTagging
- PutBucketTagging

Enabling Application Access to Object Storage

To enable application access from Amazon S3 to Object Storage, you need to set up access to Oracle Cloud Infrastructure and modify your application.

Setting up access to Oracle Cloud Infrastructure:

- Create a Oracle Cloud Infrastructure tenant. See Signing Up for Oracle Cloud Infrastructure.
- Create an Amazon S3 Compatibility API key. An Amazon S3 Compatibility API key consists of an Access Key/Secret Key pair.

Modifying your application:

- Configure a new endpoint for the application. For example: `mynamespace.compat.objectstorage.us-phoenix-1.oraclecloud.com`.
- Set the target region as one of the Oracle Cloud Infrastructure regions.



Note

If you cannot set the target region name to the correct Oracle Cloud Infrastructure region name, you must either set the region to `us-east-1` or leave it blank. Using this configuration, you can only use the Amazon S3 Compatibility API in your Oracle Cloud Infrastructure home region.

- Configure the application to use the Amazon S3 Compatibility API key.
- Make sure that you aren't using the virtual-hosted style URL, which is not supported.

At this point, you can start accessing Object Storage.

Supported Amazon S3 Clients

Qualified support is provided for the AWS SDK for Java client.

Here is an example of configuring the AWS SDK for Java (https://aws.amazon.com/documentation/sdk-for-java/) to talk to Object Storage's Amazon S3-compatible endpoints:

```
// Get S3 credentials from the console and put them here
AWSCredentialsProvider credentials = new AWSStaticCredentialsProvider(new BasicAWSCredentials(
 "ocid1.credential.oc1..anEXAMPLE",
 "anEXAMPLE="));

// The name of your tenancy
String tenancy = "tenancy";

// The region to connect to
String region = "us-ashburn-1";

// Create an S3 client pointing at the region
String endpoint = String.format("%s.compat.objectstorage.%s.oraclecloud.com",tenancy,region);
AwsClientBuilder.EndpointConfiguration endpointConfiguration = new
AwsClientBuilder.EndpointConfiguration(endpoint, region);
AmazonS3 client = AmazonS3Client.builder()
 .standard()
 .withCredentials(credentials)
 .withEndpointConfiguration(endpointConfiguration)
 .disableChunkedEncoding()
 .enablePathStyleAccess()
 .build();
```

Amazon S3 Compatibility API Requirement

Before you can use the Amazon S3 Compatibility API, you must create an Amazon S3 Compatibility API key.

After you've generated the necessary key, you can use the Amazon S3 Compatibility API to access Object Storage in Oracle Cloud Infrastructure. For more information, see the API Reference.

CHAPTER 21 Registry

This chapter explains how to store, share, and manage development artifacts like Docker images in an Oracle-managed registry.

Overview of Registry

Oracle Cloud Infrastructure Registry is an Oracle-managed registry that enables you to simplify your development to production workflow. Oracle Cloud Infrastructure Registry makes it easy for you as a developer to store, share, and manage development artifacts like Docker images. And the highly available and scalable architecture of Oracle Cloud Infrastructure ensures you can reliably deploy your applications. So you don't have to worry about operational issues, or scaling the underlying infrastructure.

You can use Oracle Cloud Infrastructure Registry as a private Docker registry for internal use, pushing and pulling Docker images to and from the Registry using the Docker V2 API and the standard Docker command line interface (CLI). You can also use Oracle Cloud Infrastructure Registry as a public Docker registry, enabling any user with internet access and knowledge of the appropriate URL to pull images from public repositories in Oracle Cloud Infrastructure Registry.

Oracle Cloud Infrastructure Registry supports private access from other Oracle Cloud Infrastructure resources in a virtual cloud network (VCN) in the same region through a service gateway. Setting up and using a service gateway on a VCN lets resources (such as worker nodes in clusters managed by Container Engine for Kubernetes) access Oracle Cloud Infrastructure services such as Oracle Cloud Infrastructure Registry without exposing them to the public internet. No internet gateway is required and resources can be in a private subnet and use only private IP addresses. For more information, see Access to Oracle Services: Service Gateway.

Oracle Cloud Infrastructure Registry is integrated with IAM, which provides easy authentication with native Oracle Cloud Infrastructure identity.

For an introductory tutorial, see Pushing an Image to Oracle Cloud Infrastructure Registry.

Ways to Access Oracle Cloud Infrastructure

You can access Oracle Cloud Infrastructure using the Console (a browser-based interface) or the REST API. Instructions for the Console and API are included in topics throughout this guide. For a list of available SDKs, see Software Development Kits and Command Line Interface.

To access the Console, you must use a supported browser. Oracle Cloud Infrastructure supports the latest desktop versions of Google Chrome, Microsoft Edge, Internet Explorer 11, Safari, Firefox, and Firefox ESR. Note that private browsing mode is not supported for Firefox, Internet Explorer, or Edge. Mobile browsers are not supported.

For general information about using the API, see REST APIs.

Resource Identifiers

Most types of Oracle Cloud Infrastructure resources have a unique, Oracle-assigned identifier called an Oracle Cloud ID (OCID). For information about the OCID format and other ways to identify your resources, see Resource Identifiers.

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

Registry Capabilities and Limits

In each region that is enabled for your tenancy, you can create up to 500 repositories in Oracle Cloud Infrastructure Registry. Each repository can hold up to 500 images. See Service Limits.

Required IAM Service Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

If you're new to policies, see Getting Started with Policies and Common Policies.

For more details about policies for Oracle Cloud Infrastructure Registry, see:

- Policies to Control Repository Access
- Details for Registry

Preparing for Registry

Before you can push and pull Docker images to and from Oracle Cloud Infrastructure Registry:

- You must have access to an Oracle Cloud Infrastructure tenancy. The tenancy must be subscribed to one or more of the regions in which Registry is available (see Availability by Region Name and Region Code).
- You must have access to the Docker CLI (for example, to push and pull images on a local machine, you'll need to have installed Docker on the local machine).
- You must either belong to a group to which a policy grants the appropriate permissions, or belong to the tenancy's Administrators group. See Policies to Control Repository Access.

- You must have an Oracle Cloud Infrastructure auth token. If you don't have an auth token already, see Getting an Auth Token.

Availability by Region Name and Region Code

Registry is available in the following regions. Note that you have to use the region code in some commands. In some cases, you might have to use shortened versions of availability domain names.

Region Name	Region Code	Shortened Availability Domain Names
Ashburn	iad	• US-ASHBURN-AD-1 • US-ASHBURN-AD-2 • US-ASHBURN-AD-3
Frankfurt	fra	• EU-FRANKFURT-1-AD-1 • EU-FRANKFURT-1-AD-2 • EU-FRANKFURT-1-AD-3
London	lhr	• UK-LONDON-1-AD-1 • UK-LONDON-1-AD-2 • UK-LONDON-1-AD-3
Phoenix	phx	• PHX-AD-1 • PHX-AD-2 • PHX-AD-3
Seoul	icn	• AP-SEOUL-1-AD-1
Tokyo	nrt	• AP-TOKYO-1-AD-1
Toronto	yyz	• CA-TORONTO-1-AD-1

About Images

You can store, share, and manage Docker images in Oracle Cloud Infrastructure Registry. A Docker image is a read-only template with instructions for creating a Docker container. A Docker image holds the application that you want Docker to run as a container, along with any dependencies. To create a Docker image, you first create a Dockerfile to describe that application. You then build the Docker image from the Dockerfile. Having created a Docker image, you store it in a Docker registry such as Oracle Cloud Infrastructure Registry.

About Repositories

Related images in Oracle Cloud Infrastructure Registry can be grouped into meaningfully named repositories for convenience.

Repositories can be private or public. Any user with internet access and knowledge of the appropriate URL can pull images from a public repository in Oracle Cloud Infrastructure Registry.

You must belong to the tenancy's Administrators group or have been granted the REPOSITORY_MANAGE permission to:

- create a new public repository
- change an existing repository into a public repository
- change an existing public repository into a private repository

If you make a repository private, you (along with users belonging to the tenancy's Administrators group) will be able to perform any operation on the repository. You can use identity policies to allow other users to perform other operations on repositories (both public and private) that you create.

Typically, the images in a repository are all different versions of the same source image (for example 'acme-web-app'), with each version identified by a tag (for example, 'acme-web-app:4.6.3').

For example, for convenience you might want to group together multiple versions of the acme-web-app image in the acme-dev tenancy in the Ashburn region into a repository called

project01. You do this by including the name of the repository in the image name when you push the image, in the format `<region-code>.ocir.io/<tenancy-name>/<repo-name>/<image-name>:<tag>`. For example, `iad.ocir.io/acme-dev/project01/acme-web-app:4.6.3`. Subsequently, when you use the `docker push` command, the presence of the repository in the image's name ensures the image is pushed to the intended repository.

If you push an image that includes a repository in the image name and the repository doesn't already exist (for example, `iad.ocir.io/acme-dev/project02/acme-web-app:7.5.2`), a new private repository is created in Oracle Cloud Infrastructure Registry.

If you push an image that doesn't explicitly include a repository in the image name (for example, `iad.ocir.io/acme-dev/acme-web-app:7.5.2`), the image's name (`acme-web-app`) is used as the name of a private repository.

Alternatively, you can use the Console to create an empty repository and give it a name. If you belong to the tenancy's Administrators group or have been granted the REPOSITORY_ MANAGE permission, you can also specify whether the repository is to be private or public. Any images you subsequently push to Oracle Cloud Infrastructure Registry that include the repository in the image name are pushed to that repository.

Creating a Repository

Using the Console, you can create an empty repository in Oracle Cloud Infrastructure Registry and give it a name. Any images you subsequently push to the registry that include the repository in the image name are grouped into that repository.

Having created the new repository, you can push an image to the repository using the Docker CLI (see Pushing Images Using the Docker CLI).

Note that although creating an empty repository can be a convenient placeholder, it is not strictly necessary. If you push an image that includes a repository in the image name and the repository doesn't already exist (for example, `iad.ocir.io/acme-dev/project02/acme-web-app:7.5.2`), a new repository is created automatically. And if you push an image that doesn't include a repository in the image name (for example, `iad.ocir.io/acme-dev/acme-web-app:7.5.2`), the image's name (`acme-web-app`) is used as the name of the repository.

Using the Console

To create a repository in Oracle Cloud Infrastructure Registry:

1. In the Console, open the navigation menu. Under **Solutions, Platform and Edge**, go to **Developer Services** and click **Registry**.
2. Choose the region in which to create the repository.
3. Click **Create Repository**.
4. In the **Add Repository** dialog box, specify details for the new repository:
    - **Repository Name:** A name of your choice for the new repository. Avoid entering confidential information.
    - **Public:** Whether the new repository will be a public repository or a private repository. You can only make the new repository public if you belong to the tenancy's Administrators group or have been granted the REPOSITORY_MANAGE permission. If you make the new repository public, any user with internet access and knowledge of the appropriate URL will be able to pull images from the repository. If you make the repository private, you (along with users belonging to the tenancy's Administrators group) will be able to perform any operation on the repository.
5. Click **Submit**.

Pushing Images Using the Docker CLI

You use the Docker CLI to push images to Oracle Cloud Infrastructure Registry.

To push an image, you first use the `docker tag` command to create a copy of the local source image as a new image (the new image is actually just a reference to the existing source image). As a name for the new image, you specify the fully qualified path to the target location in Oracle Cloud Registry where you want to push the image, optionally including the name of a repository.

For example, assume you have a local image named acme-web-app:latest. Let's say you want to push this image to Oracle Cloud Infrastructure Registry with a name of acme-web-app:version2.0.test into a repository called project01 in the Ashburn region of the acme-dev

tenancy. When you use the `docker tag` command, you'd name the new image with the fully qualified path to its destination. So in this case, you'd name the new image `iad.ocir.io/acme-dev/project01/acme-web-app:version2.0.test`. Subsequently, when you use the `docker push` command, the image's name ensures it is pushed to the correct destination.

Your permissions control the images you can push to Oracle Cloud Infrastructure Registry. You can push images to repositories you've created, and to repositories that the groups to which you belong have been granted access by appropriate identity policies. If you belong to the Administrators group, you can push images to any repository in the tenancy.

To push images to Oracle Cloud Infrastructure Registry using the Docker CLI:

1. If you already have an auth token, go to the next step. Otherwise:
    a. In the top-right corner of the Console, open the **User** menu () and then click **User Settings** to view the details.
    b. On the **Auth Tokens** page, click **Generate Token**.
    c. Enter a friendly description for the auth token. Avoid entering confidential information.
    d. Click **Generate Token**. The new auth token is displayed.
    e. Copy the auth token immediately to a secure location from where you can retrieve it later, because you won't see the auth token again in the Console.
    f. Close the Generate Token dialog.
2. In a terminal window on the client machine running Docker, log in to Oracle Cloud Infrastructure Registry by entering `docker login <region-code>.ocir.io`, where `<region-code>` corresponds to the code for the Oracle Cloud Infrastructure Registry region you're using. For example, `docker login iad.ocir.io`. See Availability by Region Name and Region Code for the list of region codes.
3. When prompted, enter your username in the format `<tenancy_name>/<username>`. For example, `acme-dev/jdoe@acme.com`. If your tenancy is federated with Oracle Identity Cloud Service, use the format `<tenancy-name>/oracleidentitycloudservice/<username>`.

4. When prompted, enter the auth token you copied earlier.
5. Locate the image on the client machine that you want to push:
    a. In a terminal window on your client machine, enter `docker images` to list the available images.
    For example:

```
$ docker images
REPOSITORY        TAG               IMAGE ID      CREATED       SIZE
acme-web-app      latest            8e0506e14874  2 hours ago   162.6 MB
acme-web-app      version1.0        7d9495d03763  2 hours ago   162.6 MB
<none>            <none>            6ebd328f833d  5 hours ago   162.6 MB
hello-world       latest            80b84820d442  5 weeks ago   890 B
```

    b. Find the image on the client machine that you want to push to Oracle Cloud Infrastructure Registry.
    In the output of the `docker images` command, look for the specific image that you want to push. You'll need to uniquely identify this image later, in one of the following ways:
        - using its id
        - using its name and tag, separated by a colon

    For example, you might have an image named `acme-web-app` on the client machine. In the output of the `docker images` command, look for the specific acme-web-app image that you want to push. You can uniquely identify that particular image in one of the following ways:
        - using its id (for example, `8e0506e14874`)
        - using its name and tag, separated by a colon (for example `acme-web-app:latest`)

    c. Give a tag to the image that you're going to push to Oracle Cloud Infrastructure Registry by entering:

```
docker tag <image-identifier> <target-tag>
```

where:

- `<image-identifier>` uniquely identifies the image, either using the image's id (for example, `8e0506e14874`), or the image's name and tag separated by a colon (for example, `acme-web-app:latest`).
- `<target-tag>` is in the format `<region-code>.ocir.io/<tenancy-name>/<repos-name>/<image-name>:<tag>` where:
  - `<region-code>` is the code for the Oracle Cloud Infrastructure Registry region you're using. For example, `iad`. See Availability by Region Name and Region Code for the list of region codes.
  - `ocir.io` is the Oracle Cloud Infrastructure Registry name.
  - `<tenancy-name>` is the name of the tenancy that owns the repository to which you want to push the image (for example, `acme-dev`). Note that your user must have access to the tenancy.
  - `<repo-name>` (if specified) is the name of a repository to which you want to push the image (for example, `project01`). Note that specifying a repository is optional (see About Repositories).
  - `<image-name>` is the name you want to give the image in Oracle Cloud Infrastructure Registry (for example, `acme-web-app`).
  - `<tag>` is an image tag you want to give the image in Oracle Cloud Infrastructure Registry (for example, `version2.0.test`).

For example, combining the previous examples, you might enter:

```
docker tag 8e0506e14874 iad.ocir.io/acme-dev/project01/acme-web-app:version2.0.test
```

6. Confirm that the Docker image has been correctly tagged on the client machine by entering `docker images` and verifying that the list of images includes an image with the tag you specified.
   For example:

```
$ docker images
REPOSITORY                                TAG                 IMAGE ID      CREATED
SIZE
```

```
iad.ocir.io/acme-dev/project01/acme-web-app version2.0.test       8e0506e14874  1 minute ago
162.6 MB
acme-web-app                                latest                8e0506e14874  2 hours ago
162.6 MB
acme-web-app                                version1.0            7d9495d03763  2 hours ago
162.6 MB
<none>                                      <none>                6ebd328f833d  5 hours ago
162.6 MB
hello-world                                 latest                80b84820d442  5 weeks ago   890
B
```

7. Push the Docker image from the client machine to Oracle Cloud Infrastructure Registry by entering:

```
docker push <target-tag>
```

where `<target-tag>` is in the format `<region-code>.ocir.io/<tenancy-name>/<repos-name>/<image-name>:<tag>` where:

- `<region-code>` is the code for the Oracle Cloud Infrastructure Registry region you're using. For example, `iad`. See Availability by Region Name and Region Code for the list of region codes.
- `ocir.io` is the Oracle Cloud Infrastructure Registry name.
- `<tenancy-name>` is the name of the tenancy that owns the repository to which you want to push the image (for example, `acme-dev`). Note that your user must have access to the tenancy.
- `<repo-name>` (if specified) is the name of a repository to which you want to push the image (for example, `project01`). Note that specifying a repository is optional (see About Repositories).
- `<image-name>` is the name you want to give the image in Oracle Cloud Infrastructure Registry (for example, `acme-web-app`).
- `<tag>` is an image tag you want to give the image in Oracle Cloud Infrastructure Registry (for example, `version2.0.test`).

For example:

```
docker push iad.ocir.io/acme-dev/project01/acme-web-app:version2.0.test
```

Pulling Images Using the Docker CLI

You use the Docker CLI to pull images from Oracle Cloud Infrastructure Registry.

Your permissions control the images you can pull from Oracle Cloud Infrastructure Registry. You can pull images from repositories you've created, from public repositories, and from repositories that the groups to which you belong have been granted access by identity policies. If you belong to the Administrators group, you can pull images from any repository in the tenancy.

To pull images from Oracle Cloud Infrastructure Registry using the Docker CLI:

1. If you already have an auth token, go to the next step. Otherwise:
    a. In the top-right corner of the Console, open the **User** menu () and then click **User Settings** to view the details.
    b. On the **Auth Tokens** page, click **Generate Token**.
    c. Enter a friendly description for the auth token. Avoid entering confidential information.
    d. Click **Generate Token**. The new auth token is displayed.
    e. Copy the auth token immediately to a secure location from where you can retrieve it later, because you won't see the auth token again in the Console.
    f. Close the Generate Token dialog.
2. In a terminal window on the client machine running Docker, log in to Oracle Cloud Infrastructure Registry by entering `docker login <region-code>.ocir.io`, where `<region-code>` corresponds to the code for the Oracle Cloud Infrastructure Registry region you're using. For example, `docker login iad.ocir.io`. See Availability by Region Name and Region Code for the list of region codes.
3. When prompted, enter your username in the format `<tenancy_name>/<username>`. For example, `acme-dev/jdoe@acme.com`. If your tenancy is federated with Oracle Identity Cloud Service, use the format `<tenancy-name>/oracleidentitycloudservice/<username>`.
4. When prompted, enter the auth token you copied earlier.

5. Pull the Docker image from Oracle Cloud Infrastructure Registry to the client machine by entering:

```
docker pull <region-code>.ocir.io/<tenancy-name>/<repos-name>/<image-name>:<tag>
```

   where:

   - `<region-code>` is the code for the Oracle Cloud Infrastructure Registry region you're using. For example, `iad`. See Availability by Region Name and Region Code for the list of region codes.
   - `ocir.io` is the Oracle Cloud Infrastructure Registry name.
   - `<tenancy-name>` is the name of the tenancy that owns the repository from which you want to pull the image (for example, `acme-dev`). Note that your user must have access to the tenancy.
   - `<repo-name>` (optional) is the name of a repository from which you want to pull the image (for example, `project01`). Note that your user must have access to the repository. Omit this argument if the image does not exist within a repository (see About Repositories).
   - `<image-name>` is the name of the image that you want to pull from Oracle Cloud Infrastructure Registry (for example, `acme-web-app`)
   - `<tag>` is the tag of the image that you want to pull from Oracle Cloud Infrastructure Registry (for example, `version2.0.test`)

   For example:

```
docker pull iad.ocir.io/acme-dev/project01/acme-web-app:version2.0.test
```

   Note that if you don't specify a `<tag>` in the `docker pull` command, Docker pulls the image that has the `latest` tag.

6. Confirm that the image has been pulled from Oracle Cloud Infrastructure Registry by entering `docker images` and verifying that the list of images on the client machine now includes the image you just pulled.

   For example:

```
$ docker images
REPOSITORY                                TAG                 IMAGE ID            CREATED
```

```
SIZE
iad.ocir.io/acme-dev/project01/acme-web-app version2.0.test     8e0506e14874      1 minute ago
162.6 MB
acme-web-app                                latest              8e0506e14874      2 hours ago
162.6 MB
acme-web-app                                version1.0          7d9495d03763      2 hours ago
162.6 MB
<none>                                      <none>              6ebd328f833d      5 hours ago
162.6 MB
hello-world                                 latest              80b84820d442      5 weeks ago
890 B
```

Pulling Images from Registry during Kubernetes Deployment

During the deployment of an application to a Kubernetes cluster, you'll typically want one or more images to be pulled from a Docker registry. In the application's manifest file you specify the images to pull, the registry to pull them from, and the credentials to use when pulling the images. The manifest file is commonly also referred to as a pod spec, or as a deployment.yaml file (although other filenames are allowed).

If you want the application to pull images that reside in Oracle Cloud Infrastructure Registry, you have to perform two steps:

- You have to use kubectl to create a Docker registry secret. The secret contains the Oracle Cloud Infrastructure credentials to use when pulling the image. When creating secrets, Oracle strongly recommends you use the latest version of kubectl (see the kubectl documentation).
- You have to specify the image to pull from Oracle Cloud Infrastructure Registry, including the repository location and the Docker registry secret to use, in the application's manifest file.

To create a Docker registry secret:

1. If you haven't already done so, download the cluster's kubeconfig configuration file and set the KUBECONFIG environment variable to point to the file. See Downloading a

kubeconfig File to Enable Cluster Access.

2. In a terminal window, enter:

```
$ kubectl create secret docker-registry <secret-name> --docker-server=<region-code>.ocir.io --docker-username='<tenancy-name>/<oci-username>' --docker-password='<oci-auth-token>' --docker-email='<email-address>'
```

where:

- `<secret-name>` is a name of your choice, that you will use in the manifest file to refer to the secret . For example, `ocirsecret`
- `<region-code>` is the code for the Oracle Cloud Infrastructure Registry region you're using. For example, `iad`. See Availability by Region Name and Region Code for the list of region codes.
- `ocir.io` is the Oracle Cloud Infrastructure Registry name.
- `<tenancy-name>` is the tenancy containing the repository from which the application is to pull the image. For example, `acme-dev`
- `<oci-username>` is the username to use when pulling the image. The username must have access to the tenancy specified by `<tenancy-name>`. For example, `jdoe@acme.com` . If your tenancy is federated with Oracle Identity Cloud Service, use the format `oracleidentitycloudservice/<oci-username>`
- `<oci-auth-token>` is the auth token of the user specified by `<oci-username>`. For example, `k]j64r{1sJSSF-;)K8`
- `<email-address>` is an email address. An email address is required, but it doesn't matter what you specify. For example, `jdoe@acme.com`

Note the use of single quotes around strings containing special characters.
For example, combining the previous examples, you might enter:

```
$ kubectl create secret docker-registry ocirsecret --docker-server=phx.ocir.io --docker-username='acme-dev/jdoe@acme.com' --docker-password='k]j64r{1sJSSF-;)K8' --docker-email='jdoe@acme.com'
```

Having created the Docker secret, you can now refer to it in the application manifest file.

To specify the image to pull from Oracle Cloud Infrastructure Registry, along with the Docker secret to use, during deployment of an application to a cluster:

1. Open the application's manifest file in a text editor.
2. Add the following sections to the manifest file:
    a. Add a `containers` section that specifies the name and location of the container you want to pull from Oracle Cloud Infrastructure Registry, along with other deployment details.
    b. Add an `imagePullSecrets` section to the manifest file that specifies the name of the Docker secret you created to access the Oracle Cloud Infrastructure Registry.

    Here's an example of what the manifest might look like when you've added the `containers` and `imagePullSecrets` sections:

```
apiVersion: v1
kind: Pod
metadata:
  name: ngnix-image
spec:
  containers:
    - name: ngnix
      image: phx.ocir.io/acme-dev/project01/ngnix-lb:latest
      imagePullPolicy: Always
      ports:
      - name: nginx
        containerPort: 8080
        protocol: TCP
  imagePullSecrets:
    - name: ocirsecret
```

3. Save and close the manifest file.

Viewing Images and Image Details

To make sure you pull the correct image or to identify images that you no longer need, you can find out detailed information about the images in Oracle Cloud Infrastructure Registry.

Your permissions control the images in Oracle Cloud Infrastructure Registry that you can view information about. You can view information about images in repositories you've created, and

in repositories that the groups to which you belong have been granted access by identity policies. If you belong to the Administrators group, you can view information about images in any repository in the tenancy.

Using the Console

To view images and image details:

1. In the Console, open the navigation menu. Under **Solutions, Platform and Edge**, go to **Developer Services** and click **Registry**.
2. Choose the registry's region. You see all the repositories in the registry to which you have access.
3. Click the name of the repository that contains the image you want to see detailed information about. You see all the different images in the repository, along with the tag of each image and when it was pushed to the registry. You can sort the different images by the date they were pushed or by their tag.
4. Click the image for which you want to see detailed information. The **Summary** page shows you the size of the image, when it was pushed and by which user, and the number of times the image has been pulled. Use the options on the **Summary** page as follows:
    - Display the **Layers** tab to see the SHA message digest of each layer in the selected image.
    - Display the **Associated Tags** tab to see the full path for the image with the tag you select. Note that if you select a different tag, the summary details change accordingly.
5. (Optional) If you want to pull an image, click the Download button beside the image name and copy the command shown. For example, `docker pull iad.ocir.io/acme-dev/project01/acme-web-app:version2.0.test`. See Pulling Images Using the Docker CLI.

Deleting an Image

When you no longer need an old image or you simply want to clean up the list of image tags in a repository, you can delete images from Oracle Cloud Infrastructure Registry.

Your permissions control the images in Oracle Cloud Infrastructure Registry that you can delete. You can delete images from repositories you've created, and from repositories that the groups to which you belong have been granted access by identity policies. If you belong to the Administrators group, you can delete images from any repository in the tenancy.

Note that as well deleting individual images as described below, you can set up image retention policies to delete images automatically based on selection criteria you specify (see Retaining and Deleting Images Using Retention Policies).

Using the Console

To delete an image from Oracle Cloud Infrastructure Registry:

1. In the Console, open the navigation menu. Under **Solutions, Platform and Edge**, go to **Developer Services** and click **Registry**.
2. Choose the registry's region. You see all the repositories to which you have access.
3. Click the name of the repository from which to delete the image.
4. Click the name of the image that you want to delete.
5. Click **Delete** on the **Summary** page and confirm that you want to delete the image.

The image is permanently removed from Oracle Cloud Infrastructure Registry.

Retaining and Deleting Images Using Retention Policies

You can set up image retention policies to automatically delete images that meet particular selection criteria, namely:

- images that have not been pulled for a certain number of days
- images that have not been tagged for a certain number of days

- images that have not been given particular Docker tags specified as exempt from automatic deletion

An hourly process checks images against the selection criteria, and any that meet the selection criteria are automatically deleted.

You'll often find image retention policies are a more convenient way to manage the images in a repository than manually deleting individual images (see Deleting an Image).

In each region in a tenancy, there's a global image retention policy. The global image retention policy's default selection criteria retain all images, so that no images are automatically deleted. However, you can change the global image retention policy so that images are deleted if they meet the criteria you specify. A region's global image retention policy applies to all repositories in the region, unless it is explicitly overridden by one or more custom image retention policies.

You can set up custom image retention policies to override the global image retention policy with different criteria for specific repositories in a region. Having created a custom image retention policy, you apply the custom retention policy to a repository by adding the repository to the policy. The global image retention policy no longer applies to repositories that you add to a custom retention policy.

If you have `manage` permission on the tenancy, you can:

- modify each region's own global image retention policy
- create new custom image retention policies
- modify the criteria of existing custom image retention policies
- delete custom image retention policies

If you have `manage` permission on a repository, you can:

- add the repository to a custom image retention policy
- remove the repository from a custom image retention policy

Note the following:

- Only one custom image retention policy at a time can apply to a repository. If a repository has already been added to a custom retention policy and you want to add the repository to a different custom retention policy, you have to remove the policy from the first retention policy before adding it to the second.
- When you create or update an image retention policy, the hourly process that checks images for deletion will ignore the new or updated policy for several hours. This cooling-off period enables you to refine the policy criteria to select only the images you want to delete, and thus reduces the chance of images being deleted unexpectedly. After this period, the policy is included in the hourly process and images are checked and deleted accordingly.
- The global image retention policy (and any custom image retention policies you create) are specific to a particular region. To delete images consistently in different regions in your tenancy, set up image retention policies in each region with identical selection criteria .

Using the Console to Edit the Global Image Retention Policy

Provided you have `manage` permission on the tenancy, you can edit the region's global image retention policy that applies to all repositories in a region (except for repositories that have been explicitly added to a custom image retention policy).

To edit the global image retention policy:

1. In the Console, open the navigation menu. Under **Solutions, Platform and Edge**, go to **Developer Services** and click **Registry**.
2. Choose the registry's region. You see all the repositories to which you have access.
3. Click **Settings**, and then select **Image retention policies**.
   You see the current selection criteria of the region's global image retention policy, along with any custom image retention policies that override the global image retention policy for specific repositories.
4. Click **Edit Global Policy**.
5. In the **Global Image Retention Policy** dialog, specify new criteria for the global

retention policy:

- **Delete any images that haven't been pulled in *n* days:** Select this option if you want to delete images that have not been pulled for the number of days you specify.
- **Delete any images that haven't been tagged in *n* days:** Select this option if you want to delete images that have not been tagged for the number of days you specify.
- **Exempt Tags:** If you want to prevent images from being deleted on the basis of Docker tags they've been given, specify those tags as exempt in a comma-separated list. An image that has been given one of the exempt tags will not be deleted, even if the image meets the other criteria. You can include the asterisk (*) as a wildcard to represent none, one, or more characters. For example, you might specify `latest,prod-*,*-tail,*.100.*`.

6. Click **Save Settings**.

Going forward, the criteria you entered for the region's global image retention policy will apply to all repositories in the region, except for repositories that have been explicitly added to a custom image retention policy. Images in repositories that have not been added to a custom image retention policy will be deleted from Oracle Cloud Infrastructure Registry if they meet the criteria you specified in the global image retention policy.

When you create or update an image retention policy, the hourly process that checks images for deletion will ignore the new or updated policy for several hours. This cooling-off period enables you to refine the policy criteria to select only the images you want to delete, and thus reduces the chance of images being deleted unexpectedly. After this period, the policy is included in the hourly process and images are checked and deleted accordingly.

Using the Console to Create a New Custom Image Retention Policy to Override the Global Policy

Provided you have `manage` permission on the tenancy, you can create a new custom image retention policy to override the region's global image retention policy for the repositories you specify. A custom image retention policy is specific to the region in which you create it.

To create a new custom image retention policy:

1. In the Console, open the navigation menu. Under **Solutions, Platform and Edge**, go to **Developer Services** and click **Registry**.
2. Choose the registry's region. You see all the repositories to which you have access.
3. Click **Settings**, and then select **Image retention policies**.
   You see the current selection criteria of the region's global image retention policy, along with any existing custom image retention policies that override the global image retention policy for specific repositories.
4. Click **Create Policy**.
5. In the **Create Repository Image Retention Policy** dialog, specify criteria for the new retention policy:
   - **Policy Name:** A name of your choice for the policy. Avoid entering confidential information.
   - **Delete any images that haven't been pulled in *n* days:** Select this option if you want to delete images that have not been pulled for the number of days you specify.
   - **Delete any images that haven't been tagged in *n* days:** Select this option if you want to delete images that have not been tagged for the number of days you specify.
   - **Exempt Tags:** If you want to prevent images from being deleted on the basis of Docker tags they've been given, specify those tags as exempt in a comma-separated list. An image that has been given one of the exempt tags will not be deleted, even if the image meets the other criteria. You can include the asterisk

(*) as a wildcard to represent none, one, or more characters. For example, you might specify `latest,prod-*,*-tail,*.100.*`.

6. Click **Save Settings**.

You can now add repositories to the new custom retention policy.

Using the Console to Remove a Repository from a Custom Image Retention Policy

Provided you have `manage` permission on a repository, you can remove a repository from a custom image retention policy to which it was previously added.

You might want to remove the repository from a custom image retention policy:

- if you want the region's global image retention policy to apply to the repository
- if you want a different custom image retention policy to apply to the repository (only one custom image retention policy at a time can apply to a repository)

To remove a repository from a custom image retention policy:

1. In the Console, open the navigation menu. Under **Solutions, Platform and Edge**, go to **Developer Services** and click **Registry**.
2. Choose the registry's region. You see all the repositories to which you have access.
3. Click **Settings**, and then select **Image retention policies**.
   You see the current selection criteria of the region's global image retention policy, along with any existing custom image retention policies that override the global image retention policy for specific repositories.
4. Locate the custom image retention policy to which the repository has been added.
5. Click the delete icon beside the repository name to remove it from the custom image retention policy.

Going forward, the region's global image retention policy will apply to the repository (unless you add the repository to a different custom image retention policy). The images in the

repository will be deleted from Oracle Cloud Infrastructure Registry if they meet the criteria specified in the global image retention policy.

When you create or update an image retention policy, the hourly process that checks images for deletion will ignore the new or updated policy for several hours. This cooling-off period enables you to refine the policy criteria to select only the images you want to delete, and thus reduces the chance of images being deleted unexpectedly. After this period, the policy is included in the hourly process and images are checked and deleted accordingly.

Using the Console to Add a Repository to a Custom Image Retention Policy

Provided you have `manage` permission on a repository, you can add a repository to an existing custom image retention policy.

Note that if a custom image retention policy already applies to the repository, you'll have to remove the repository from the current policy before adding it to a different policy. Note also that a custom image retention policy is specific to the region in which it was created.

To add a repository to an existing custom image retention policy:

1. In the Console, open the navigation menu. Under **Solutions, Platform and Edge**, go to **Developer Services** and click **Registry**.
2. Choose the registry's region. You see all the repositories to which you have access.
3. Click **Settings**, and then select **Image retention policies**.
   You see the current selection criteria of the region's global image retention policy, along with the custom image retention policies that have been defined to override the global image retention policy for specific repositories.
4. Locate the custom image retention policy to which you want to add the repository.
5. Click **Add Repository** and select from the list the repository you want to add to the custom image retention policy.
   Note that the repository list includes all repositories in the region, regardless of whether you have permission to add them to a retention policy. You can only add a repository to a retention policy if you have `manage` permission on that repository,

If a repository in the list has a policy name beside it, the repository has already been added to a policy. Before you can add the repository to a different policy, you'll have to remove it from the first policy.

Going forward, the custom retention policy to which you added the repository will override the region's global image retention policy. The images in the repository will be deleted from Oracle Cloud Infrastructure Registry if they meet the criteria specified in the custom retention policy.

When you create or update an image retention policy, the hourly process that checks images for deletion will ignore the new or updated policy for several hours. This cooling-off period enables you to refine the policy criteria to select only the images you want to delete, and thus reduces the chance of images being deleted unexpectedly. After this period, the policy is included in the hourly process and images are checked and deleted accordingly.

Deleting a Repository

There is a limit to the number of repositories you can have in any given region in a tenancy. So when you no longer need a repository, it makes sense to delete it from Oracle Cloud Infrastructure Registry.

Your permissions control the repositories in Oracle Cloud Infrastructure Registry that you can delete. You can delete repositories you've created, and from repositories that the groups to which you belong have been granted access by identity policies. If you belong to the Administrators group, you can delete any repository in the tenancy.

Using the Console

To delete a repository from Oracle Cloud Infrastructure Registry:

1. In the Console, open the navigation menu. Under **Solutions, Platform and Edge**, go to **Developer Services** and click **Registry**.
2. Choose the registry's region. You see all the repositories in the registry to which you have access.

3. Click the name of the repository that you want to delete.
4. Click **Delete** on the **Summary** page and confirm that you want to delete the repository.

The repository is permanently removed from Oracle Cloud Infrastructure Registry.

Getting an Auth Token

Before you can push and pull Docker images to and from Oracle Cloud Infrastructure Registry, you must already have an Oracle Cloud Infrastructure username and an auth token. If you haven't got an auth token, or you've forgotten it, or you're not sure, you can create a new auth token. You only see the auth token string when you create it, so be sure to copy the auth token to a secure location immediately.

Tip: Each user can have up to two auth tokens at a time. So if you do lose or forget the auth token, you can always create a second auth token.

To create a new auth token:

1. In the top-right corner of the Console, open the **User** menu () and then click **User Settings** to view the details.
2. On the **Auth Tokens** page, click **Generate Token**.
3. Enter a friendly description for the auth token. Avoid entering confidential information.
4. Click **Generate Token**. The new auth token is displayed.
5. Copy the auth token immediately to a secure location from where you can retrieve it later, because you won't see the auth token again in the Console.
6. Close the Generate Token dialog.

Policies to Control Repository Access

You have fine-grained control over the operations that users are allowed to perform on repositories in Oracle Cloud Infrastructure Registry.

A user's permissions to access repositories comes from the groups to which they belong. The permissions for a group are defined by identity policies. Policies define which actions the

members of a group can perform. Users access repositories and perform operations based on the policies set for the groups they are members of. Identity policies to control repository access must be set at the tenancy level. See Details for Registry .

Before you can control access to repositories, you must have already created users and already placed them in appropriate groups (see Managing Users and Managing Groups). You can then create policies and policy statements to control repository access (see Managing Policies).

Note that users in the tenancy's Administrators group can perform any operation on any repository in Oracle Cloud Infrastructure Registry that belongs to the tenancy.

Common Policies



Note

The policies in this section use example group names, as follows:

- acme-viewers: A group that you want to limit to seeing a list of repositories in the tenancy.
- acme-pullers: A group that you want to limit to pulling images.
- acme-pushers: A group that you want to allow to push and pull images.
- acme-managers: A group that you want to allow to push and pull images, delete repositories, and edit repository metadata (for example, to make a private repository public).

Make sure to replace the example group names with your own group names.

Enable users to view a list of all the repositories belonging to the tenancy

Type of access: Ability to see a list of all repositories in Oracle Cloud Infrastructure Registry belonging to the tenancy. Users will not be able to:

- view the images or layers in a repository
- push or pull images from or to a repository

Note that there is currently no way to restrict the repositories shown on the Registry page in the Console.

Where to create the policy: In the tenancy.

```
Allow group acme-viewers to inspect repos in tenancy
```

Enable users to pull images from any repository belonging to the tenancy

Type of access: Ability to pull images (layers and manifests) from any repository in Oracle Cloud Infrastructure Registry that belongs to the tenancy.

Where to create the policy: In the tenancy.

```
Allow group acme-pullers to read repos in tenancy
```

Enable users to pull images from specific repositories

Type of access: Ability to pull images (layers and manifests) from any repository in Oracle Cloud Infrastructure Registry that belongs to the tenancy and that has a name starting with "acme-web-app".

Where to create the policy: In the tenancy.

```
Allow group acme-pullers to read repos in tenancy where all { target.repo.name=/acme-web-app*/ }
```

Enable users to push images to any repositories (and create new repositories if necessary)

Type of access: Ability to push images (layers and manifests) to any repository in Oracle Cloud Infrastructure Registry that belongs to the tenancy. If a repository with the same name as the image doesn't exist yet, the `REPOSITORY_CREATE` permission ensures users are able to create the repository when they push the image.

Where to create the policy: In the tenancy.

```
Allow group acme-pushers to use repos in tenancy
```

```
Allow group acme-pushers to manage repos in tenancy where ANY {request.permission = 'REPOSITORY_
CREATE', request.permission = 'REPOSITORY_UPDATE'}
```

Enable managers to perform any operation on any repository belonging to the tenancy

Type of access: Ability to perform any operation on any repository in Oracle Cloud Infrastructure Registry that belongs to the tenancy, including:

- pull an image from any repository
- push an image to any repository
- create a new repository (either an empty repository, or when pushing an image for which no repository exists yet)
- delete a repository
- change a public repository to a private repository, or a private repository to a public repository

Where to create the policy: In the tenancy.

```
Allow group acme-managers to manage repos in tenancy
```

CHAPTER 22 Overview of Resource Manager

Resource Manager is an Oracle Cloud Infrastructure service that allows you to automate the process of provisioning your Oracle Cloud Infrastructure resources. It helps you install, configure, and manage resources using the "infrastructure-as-code" model.

Resource Manager runs as an Oracle Cloud Infrastructure service and uses Terraform to codify your infrastructure in declarative configuration files, which allows you to review and edit, version, persist, reuse, and share them across teams. You can then use Resource Manager to provision Oracle Cloud Infrastructure resources using your Terraform configurations.



Note

For more information about the Oracle Cloud Infrastructure Terraform provider, see Terraform Provider. For a general introduction to Terraform and the "infrastructure-as-code" model, see Terraform: Write, Plan, and Create Infrastructure as Code.

While you can install and run Terraform locally and use Oracle Terraform modules to do specific tasks, using Resource Manager allows you to share and manage infrastructure configurations and state files across multiple teams and platforms.

You can run Resource Manager using either the command line interface (CLI) or the Oracle Cloud Infrastructure Console. The Console provides an easy-to-use interface, while the CLI enables programmatic interaction with the Resource Manager. You can also program directly by using the Software Development Kits and Command Line Interface, or by using the Resource Manager REST APIs. Oracle provides API reference documentation in the Resource Manager API Reference.

Resource Manager integrates with Identity and Access Management (IAM), allowing you to define granular permissions using policies. It also integrates seamlessly with the Oracle Cloud Infrastructure Audit service to enable auditing for your infrastructure provisioning.

Authentication and Authorization

To use Resource Manager, you must configure the required Identity and Access Management (IAM) permissions.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Many tasks in the Resource Manager require specific permissions for the user who is performing the action. For details and examples, see:

- Details for Resource Manager for a complete list of Resource Manager permissions
- Managing Stacks and Jobs for example policies for Resource Manager

Permissions are scoped to the tenancy compartment. When your tenancy is first provisioned, a root compartment is created automatically. The administrator can create more compartments and control access to each one and its resources by creating IAM policies. These policies specify which actions each group can take on the resources in each compartment. For information about IAM permissions, see Overview of Oracle Cloud Infrastructure Identity and Access Management. See also How Policies Work.

Tagging Resources

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

Key Concepts

Following are brief descriptions of key concepts and the main components of Resource Manager.

Stacks

Stacks represent definitions of groups of Oracle Cloud Infrastructure resources that you can act upon as a group. Each stack has a configuration, which is made up of one or more declarative configuration files. Stacks are attached to a specific region. However, where necessary, the resources on a given stack can be deployed across multiple regions.

Stacks reside in a compartment of your choosing inside your Oracle Cloud Infrastructure tenancy. Jobs reside in the compartment that is occupied by the stack they are associated with. Resource Manager assigns unique Oracle Cloud IDs (OCIDs) to both stacks and jobs.

Jobs

Jobs perform the actions that are defined in your configuration. Only one job at a time can run on a given stack; further, you can have only one set of Oracle Cloud Infrastructure resources on a given stack. When you have to run a job to provision a different set of resources, you must create a separate stack and use a different configuration.

The Resource Manager provides three job types:

- **Plan job**. A plan job takes your Terraform configuration, parses it, and creates an execution plan. The execution plan lists a sequence of specific actions that take place to provision your Oracle Cloud Infrastructure resources. The execution plan is handed off to the apply job, which then executes the instructions.
- **Apply job**. The apply job takes your execution plan, applies it to the associated stack, then executes the configuration's instructions. By doing so, it creates (or modifies) the stack resources. Depending on the number and type of resources specified, apply jobs can result in a long-running operation. Both Console and the CLI allow you to determine the status of job while it executes.

- **Destroy job**. To clean up the infrastructure controlled by the stack, you run a destroy job. A destroy job does not delete the stack and its resources, but instead releases the resources (terminates a Compute instance, for example). However, the stack's job history and state remain. You can monitor the status and review the results of a destroy job by inspecting the stack's log files.

In a sense, jobs represent a record of a stack's history, which you can see by reviewing the following:

- The configuration (snapshot) for a given apply job.
- The execution plan that is derived by a plan job.
- The state file for a given apply job.

Tenancy

A tenancy is a secure, isolated partition within the Oracle Cloud Infrastructure ecosystem where you can create, organize, and administer your cloud resources. The scope of a tenancy is typically an entire organization. You can subdivide your organization's resources by creating compartments that align with your business.

Compartment

Organize and control access to your cloud resources by segmenting them into compartments. A compartment is a collection of related resources such as instances, virtual cloud networks, block volumes, and so forth. Compartments also form security barriers that employ role-based access control to regulate access to your cloud resources.

Compartments create logical groupings of resources that you can align with your business. For example, you might have a human resources compartment, a finances compartment, an operations compartment, and so forth.

Region

Regions represent geographical locations where Oracle Cloud Infrastructure hosting resources are physically located. Each region contains availability domains, which are individual data centers. For more information about regions and availability domains, see Regions and Availability Domains.

Configuration

A configuration is a set of one or more Terraform configuration (.tf) files that specify the Oracle Cloud Infrastructure resources in a given stack, including resource metadata, and other important information, including data source definitions, variables declarations, and so forth.

Configurations are simple text files that describe your infrastructure using a declarative language (HashiCorp Configuration Language, or HCL). Alternatively, you can provide configurations using JSON format when your configurations must be machine readable. Configuration files that use the HCL format end with the `.tf` file extension; those using JSON format end with the `.tf.json` file extension. The HCL format is human-readable, while the JSON format is machine readable.

To see example configuration files, you can review existing Terraform provider examples. For more information, see Writing Terraform Configurations; see also Hashicorp: Configuration.

State (state files)

Essential information about the state of your resources configuration is maintained in a state (.tfstate) file, which uses the JSON format. The state file maps your stack's resources to your configuration and also maintains essential configuration metadata, such as resource dependencies. Resource Manager generates and updates state files automatically. You cannot edit the file manually.

Resource Manager supports state locking by allowing only one job at a time to run on a given stack. For more information about state files, see Hashicorp: State.

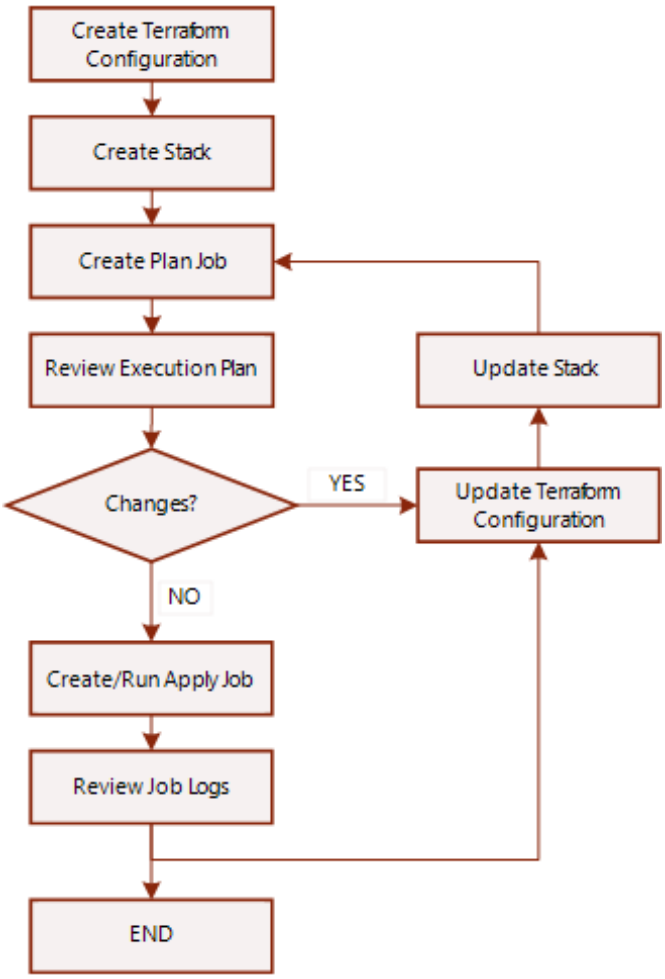
Modules

Terraform supports using modules to group related resources. Modules can be used to create lightweight and reusable abstractions, so that you can describe your infrastructure in terms of its architecture. For more information, see Creating Modules.

We recommend that all modules used by a Resource Manager stack are included locally in the configuration and referenced using a local path. To use a module for Oracle Cloud Infrastructure from the Terraform Module Registry, download the source from GitHub and include the relevant portion in a subdirectory in your zip file, then reference the module using a local path. For more information, see Local Paths.

Generalized Workflow

The following image represents a generalized view of the Resource Manager workflow.



Each step in the generalized workflow is discussed in detail the topics Using Resource Manager with the CLI and Using Resource Manager in the Console.

1. Create a Terraform configuration.
2. Create a stack.
3. Create a plan job, which produces an execution plan.
4. Review the execution plan.

5. If changes are needed in the execution plan, update the configuration and recreate the plan job.
6. Create and run the apply job to provision resources.
7. Review state file and log files, as needed.
8. You can optionally reapply your configuration, with or without making changes.
9. Optionally, to release the resources running on a stack, run a destroy job .

Using Resource Manager in the Console

You can use the Oracle Resource Manager on the command line using the CLI or by using the Oracle Cloud Infrastructure Console. This topic walks through completing Resource Manager tasks using the Console. For more information on the CLI workflow, see Using Resource Manager with the CLI.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

For more information on getting started using the Oracle Cloud Infrastructure Console, see Signing In to the Console and Understanding Compartments.

For a high-level summary of the Resource Manager, see Overview of Resource Manager.

About Terraform Configurations

These guidelines for using the Console with Resource Manager assume that you are using a Terraform configuration.

Configurations set variables and describe your infrastructure in declarative code. The configuration files are bundled into a .zip file then uploaded to Resource Manager.

A Terraform configuration is a set of files that:

- Defines the Terraform provider
- Specifies which resources are provisioned
- Defines variables
- Provides specific instructions for the apply job to follow when provisioning resources in the stack

For more information about configurations, see Terraform Configuration and Writing Terraform Configurations.

Resource Manager supports the native Terraform behavior for handling variables. You can include a `terraform.tfvars` file and files with the `.auto.tfvars` extension in the configuration .zip file. For more information about defining variables, see Define Variables and Input Variables.



Note

Resource Manager requires the following structure for the configuration:

- Configuration .zip files should not contain Terraform state files.
- You should not set the API private key in the configuration.
- Resource Manager will return an error if the .zip file is empty, contains no .tf files in the working directory, contains a .terraform directory, or is corrupted.

Writing the configuration is the most important part of using Resource Manager to provision your Oracle Cloud Infrastructure tenancy. After you create a configuration, you can version, persist, and share that configuration.

Declaring the Terraform Provider

When using Resource Manager, the `region` field in the `provider "oci"` block is the only required field. All other fields, such as `userid` or `fingerprint`, are optional.

Example

The following example shows a configuration that is contained in a single file. This very basic sample defines just one Terraform provider, one Oracle Cloud Infrastructure resource, and declares a set of variables.

```
variable "compartment_ocid" {}
variable "region" {}

provider "oci" {
  region = "${var.region}"
}

resource "oci_core_virtual_network" "vcn1" {
  cidr_block = "10.0.0.0/16"
  dns_label = "vcn1"
  compartment_id = "${var.compartment_ocid}"
  display_name = "vcn1"
}
```

More often, Terraform configurations consist of two or more files bundled together and uploaded in a .zip file. To see more complex, multi-file examples of Terraform configurations, explore the examples at the Oracle Cloud Infrastructure GitHub: terraform-provider-oci/docs/examples.

Access the Resource Manager

After you sign in to the Console, you can access the Resource Manager by opening the navigation menu in the upper left corner of the Console. Select **Resource Manager**, then select **Stacks** or **Jobs.** Resource Manager defaults to **Stacks**.

In addition to the navigation menu, there are also the following icons and menus at the top of the Console:

- Search box
- Region selection drop-down for selecting your region
- Question-mark icon for accessing Console documentation and support
- User icon for accessing user settings and tenancy information

Click the user icon. From the user information drop-down, select **Tenancy**. The page displays important information and provides access to tenancy settings:

- Tenancy name, tenancy OCID, and region. For more information, see Setting Up Your Tenancy. See also Managing the Tenancy.
- Displays the audit retention period and provides an **Edit Audit Retention Policy** button. See Setting Audit Log Retention Period for more information.
- Displays the object storage settings and provides a button to edit these settings. See Overview of Object Storage for more information.
- Provides an **Apply Tags** button, and also a **Tags** tab that lists details about existing tags. For more information, see Managing Tags and Tag Namespaces.
- Provides a link to Service Limits for the tenancy.



Note

The tenancy OCID is on the **Tenancy Details** page. Open the User menu and click **Tenancy:** *<your_tenancy_name>*. The tenancy OCID is shown under **Tenancy Information**.

Compartments

On the left of the Console, below the **List Scope** label, a drop-down labeled **Compartment** allows you to select a compartment. The list of compartments is filtered to display only those

compartments that you have access to. If a compartment that you expect to see is not listed, contact your administrator.

Compartments allow you to organize your resources and control access to compartments and the resources they contain. Each tenancy is created with a default "root" compartment. The tenancy administrator can then create additional compartments, then restrict access to each compartment and its resources using role-based access control (RBAC) provided by Identity and Access Management (IAM).

For more information about compartments, see Managing Compartments. For more information about access control, see Overview of Oracle Cloud Infrastructure Identity and Access Management.

Managing Stacks

Use stacks to organize groups of Oracle Cloud Infrastructure resources that you can then use jobs to act on as a group. Stacks and the resources they contain are scoped to compartments, so you can only act on stacks that reside in the selected compartment.



Important

Do not add your private key or other confidential information to configuration variables.

Click **Stacks** to show a table of all of the stacks in the compartment. The table lists the stack name, an optional description, the current state of the stack, and the stack's create date and time. To act on a specific stack, click the stack name to open that stack's action page.

You can click the action icon (three dots) at the right side of the job row to do the following:

- **View Stack Details** displays Stack Information and the Jobs table
- **Edit** opens a dialog to edit name, description, and variables for the stack
- **Copy OCID** copies the stack's OCID to clipboard
- **Delete** deletes the stack

You can perform several actions from the stack's action page:

Create a Stack

1. Click the **Create Stack** button, which opens a **Create Stack** dialog in which to complete the following actions.
2. From the drop-down, you can select any compartment in the tenancy in which to create the stack. A compartment from the list scope is set by default.
3. Provide a name for the new stack. If you do not provide a name, a default name is provided on the server.
4. Provide a description. Doing so is optional, but recommended.
5. Add your Resource Manager configuration (.zip) file, either by dragging and dropping it onto the dialog's control, or by clicking **Browse** and navigating to the file location. The template is a .zip file that contains one or more Terraform configuration (.tf) files, and (optionally) other supporting files.



Important

By default, Resource Manager looks for the configuration (.tf) files in the root directory of the .zip file. If they are located elsewhere, use the workingDirectory parameter to specify that location.



Warning

Do not include the **.terraform** directory in the working directory in your configuration zip file. Including **.terraform** in your working directory will cause an `invalid parameter` error when creating or updating a stack.

6. Define your stack's variables by clicking the **+Additional Variable** button in the **Variables** control, then adding key-value pairs as appropriate.
7. Click **Create** to finish.

Edit or Delete a Stack

1. Click the name of the stack you wish to edit or delete, which opens the **Stack Details** page for that stack.
2. To edit a stack, click **Edit Stack**, which allows you to edit the stack's name and description, upload a .zip file, or update variables.
3. To delete the stack, click **Delete Stack**. You receive a "confirm delete" dialog. Click **Delete**. Note that you cannot undo the delete stack operation.

Update a Stack

To update a stack, you must update the configuration, and then rerun the plan job.

1. On the **Stack** page, in the **Stack Information** tab, click **Upload New**.
2. Upload your revised configuration .zip file. The upload overwrites the existing configuration.
3. Click the **Terraform Actions** drop-down and select **Apply** to rerun the job.

Edit Variables

1. On the **Stack Details** page, under the **Resources** label, click **Variables**, which opens a dialog that displays all of your variable key-value pairs.
2. Click **Edit Variables**. The **Edit Stack** dialog opens. In this dialog, you can edit, add, or delete variables.

Managing Jobs

On the **Stack Details** page, the **Terraform Actions** drop-down provides three options: **Plan**, **Apply**, and **Destroy**. These are the three job types that you can perform on a stack.

- **Plan Job**. Terraform parses your configuration files and creates an execution plan that lists the actions taken and the resources that are created on the stack once the plan job is applied. We recommend creating a plan job before running an apply job.
- **Apply Job**. Terraform applies your configuration to the associated stack, creates the stack resources defined in this configuration, and then executes the actions that are defined in the configuration. The time required to complete an apply job depends on the number and type of cloud resources to be created.
- **Destroy Job**. Run a destroy job to tear down the resources and clean up the tenancy. You can monitor the status and review the results of a destroy job by inspecting the stack's state file, or by reviewing log files.

The Jobs table lists the job name, type, state, stack, start time and end time. You can click the action icon (three dots) at the right side of the job row to do the following:

- **Download Logs** allows you to download log files for the job that you've run on the stack.
- **Download Terraform State** allows you to download the job's state (.json) file.
- **Download Execution Plan** allows you to download the execution plan generated by the plan job (see following). The execution plan is a binary file and cannot be accessed directly. Instead, view job's log files to evaluate execution of the job.

From the **Job Details** page you can perform several actions:

View Job Details

To view the details and act on jobs, click the job name in the list provided on the **Stack Details** page. Alternatively, you can click the **Jobs** link in the left navigation area, and then click the job name on the list of jobs. Either method takes you to the **Job Details** page, which provides key information about the job, such as job type and status. It also provides access to the job's execution plan, Terraform configuration, Terraform state, and log files.

Download Terraform State File, Terraform Configuration (snapshot) and Job Logs

The **Logs** section exposes a view of the job log for the selected job. You can also download a text file of the log by clicking the **Download Logs** button and then saving the file. In cases of a long-running job, click the **Refresh** button to update the log.

The **Download Terraform State** button allows you to download the state file for the specified job.

To view the variables that were used to execute the job, click the **Variables** link on the left-hand **Resources** menu.

Run a Plan Job

Executing a plan job takes the configuration, parses it, and converts the configuration to an execution plan. The execution plan is applied when you move to the next step, running the apply job.

1. Navigate to the **Stack Details** page for the stack in which you intend to create a job.
2. Click the **Terraform Actions** drop-down and select **Plan**.
3. Give the job a name, then click **Plan**. You are returned to the **Stack Details** page and the plan job you created is now in the **Jobs** list. Notice that its state is initially "Accepted." Soon the status changes to "In Progress." For more information about the job, click the job name.

Run an Apply Job

The apply job runs the stack's execution plan.

1. Navigate to the **Stack Details** page for the stack in which you intend to run the apply job.
2. Click the **Terraform Actions** drop-down and select **Apply**.
3. Click **Apply**.

Run a Destroy Job

1. Navigate to the **Stack Details** page for the stack in which you intend to run the destroy job.
2. Click the **Terraform Actions** drop-down and select **Destroy**.

Adding Tags to Stacks, Jobs, and Tenancies

Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

Tags are key/value pairs that you can attach to resources to help you organize and track your tenancy resources. This is helpful when managing large numbers of resources across several compartments. You can apply tags to two Resource Manager resource types: stacks and jobs. You can also apply tags to the Oracle Cloud Infrastructure *tenancy* resource type.

Adding Tags to a Stack

1. Select and open the **Stack Details** page for the desired stack.
2. On the **Stack Details** page, click the **Tags** tab.

3. Click **Add Tag(s)**. This opens the tagging dialog box.
4. To create a defined tag, select a namespace from the **Tag Namespace** drop-down, then enter appropriate values in the **Tag Key** and **Value** text boxes.
5. To create a free-form tag, bypass the namesspace and enter appropriate values in the **Tag Key** and **Value** text boxes.
6. To create additional tags, click **+Additional Tag** and repeat step three or four, as appropriate.
7. Finally, click **Apply Tag(s)**.

Adding Tags to a Job

1. Select and open the **Job Details** page for the desired job.
2. On the **Job Details** page, click the **Tags** tab.
3. Click **Apply Tag(s)**. This opens the tagging dialog box.
4. To create a defined tag, select a namespace from the **Tag Namespace** drop-down, then enter appropriate values in the **Tag Key** and **Value** text boxes.
5. To create a free-form tag, bypass the namesspace and enter appropriate values in the **Tag Key** and **Value** text boxes.
6. To create additional tags, click **+Additional Tag** and repeat step three or four, as appropriate.
7. Finally, click **Apply Tag(s)**.

Adding Tags to a Tenancy

1. Open the **User** menu (icon) in the upper right corner of the Console, then select **Tenancy:***<tenancy_name>* from the drop-down.
2. Click **Apply Tag(s)**. This opens the tags dialog.
3. Select the tag namespace.

4. Enter a value for the **Tag Key**, then apply a value for the key.
5. To add more tags, click **+Additional Tag** and repeat previous steps.
6. When finished, click **Apply Tag(s)**.

Using Resource Manager with the CLI

This topic provides guidance for using the Oracle Resource Manager using the Oracle Cloud Infrastructure CLI. For guidance using Resource Manager in the Console, see Using Resource Manager in the Console.

This topic assumes that you have installed and configured the CLI. For more information on setting up the CLI, see the Quickstart and Configuration.



Warning

Oracle recommends that you avoid using string values that include confidential information in the Oracle Cloud InfrastructureConsole, API, or CLI. Do not provide user credentials in your Terraform configurations.

Prepare Your Configuration

The technology that drives the Resource Manager is Terraform, and using Terraform begins by creating a Terraform configuration. A Terraform configuration is a collection of one or more configuration (.tf) files that use declarative code to describe your infrastructure configuration. For more information about Terraform configurations, see Writing Terraform Configurations. See also Terraform: Configuration.

For the purposes of this walkthrough, we use a single Terraform configuration file named vcn.tf. It is a very basic sample file that defines just one Terraform provider, one Oracle Cloud Infrastructure resource, and declares a set of variables.

```
variable "compartment_ocid" {}
variable "region" {}

provider "oci" {
  region = "${var.region}"
}

resource "oci_core_virtual_network" "vcn1" {
  cidr_block = "10.0.0.0/16"
  dns_label = "vcn1"
  compartment_id = "${var.compartment_ocid}"
  display_name = "vcn1"
}
```

Create a Stack

Use the `stack create` command and provide the compartment OCID where you intend to create the stack. Also, reference your Terraform configuration file (`vcn.tf` in the example that follows). Optionally, provide a user-friendly name for the stack, and also add a description, as shown.

To create a stack, run the following command:

```
oci resource-manager stack create --compartment-id <compartment_OCID>
    --config-source vcn.zip --variables file://variables.json
    --display-name "My Example Stack"
    --description Tutorial to create a VCN
```

When creating a stack using the `stack create` command, the `--config-source` parameter takes the name of a single .zip file that contains one or more configuration files.

To expose the directory structure, use the optional `--working-directory` parameter to point to the root configuration file in the directory. If you do not use this parameter, the service assumes that the top-level file in the directory is the root configuration file.



Warning

Do not include the .terraform directory in the working directory in your configuration .zip file. Including .terraform in your working directory will cause an invalid parameter error when calling `stack create` or `stack update`.



Important

On Windows, be sure the .zip file and variables.json files are in the same directory from which you're running the CLI. The CLI currently has a limitation on Windows that prevents correct handling of the files if either one is in a subdirectory.

Example variables.json

Following is an example variables.json file:

```
{
  "compartment_ocid": "ocid1.compartment.oc1..example7lcdlihrvur7xq",
  "region": "us-phoenix-1"
}
```

The Oracle Cloud Infrastructure Terraform provider requires additional parameters when running Terraform locally (unless you are using instance principals). For more information on using variables in Terraform, see Input Variables. See also Input Variable Configuration.

Example stack create Response

Following is an example response to the call to `stack create`:

```
{
  "data": {
```

```
    "config-source": {
      "working-directory": null,
      "config-source-type": "ZIP_UPLOAD"
    },
    "defined-tags": {},
    "description": "Tutorial to create a VCN",
    "display-name": "My Example Stack",
    "freeform-tags": {},
    "id": "ocid1.ormstack.oc1..examplej6whb4mluogzdv4xjma",
    "lifecycle-state": "ACTIVE",
    "time-created": "2018-04-03T18:26:56.299000+00:00",
    "variables": {
      "compartment_ocid": "ocid1.compartment.oc1..example7lcdlihrvur7xq",
      "region": "us-phoenix-1"
    }
  }
}
```

List Stacks in a Compartment

To retrieve a list of the stacks in a given compartment, use the `stack list` command and provide the compartment OCID, as shown following.

```
oci resource-manager stack list --compartment-id <stack_compartment_OCID>
```

List Full Details of a Stack

To retrieve the full details of a given stack, use the `stack get` command and provide the stack's OCID, as shown here:

```
oci resource-manager stack get --stack-id <stack_OCID>
```

Delete a Stack

To delete a stack, use the `stack delete` command and provide the stack's OCID, as shown here:

```
oci resource-manager stack delete --stack-id <stack_OCID>
```



Note

Before you delete a stack, first run a destroy job on the stack because `stack delete` does not clean up the stack's resources when it deletes the stack.

Run a Plan Job

After creating a stack and uploading your Terraform configuration, you can run jobs against the stack. First, run a plan job. During a plan job, Resource Manager parses the configuration files and creates an execution plan. The execution plan allows you to review the actions that will take place when you execute the configuration.

To run a plan job, use the `job create` command and pass in the appropriate stack OCID, then specify the operation, as shown here:

```
oci resource-manager job create --operation PLAN
    --stack-id <stack_OCID>
    --display-name "Example Plan Job"
```

This operation can take several minutes. At any time, you can check the current state of the plan job by calling `job get` and providing the job OCID., as shown:

```
oci resource-manager job get --job-id <plan_job_OCID>
```

The call response includes a `lifecycle-state` value, which is `SUCCEEDED` once the job has completed. In the following example response, the lifecycle-state value is `ACCEPTED`, which means the API call was valid, but the job has not yet begun.

```
{
  "data": {
    "compartment-id": " ocid1.compartment.oc1..example3e6iindlsm5e4elqa ",
    "defined-tags": null,
    "display-name": "Example Plan Job",
    "freeform-tags": {},
    "id": "ocid1.ormjob.oc1..exampleog2iwhxafi2iilsn6d3p5",
    "lifecycle-state": "ACCEPTED",
    "operation": "PLAN",
```

```
    "stack-id": " ocid1.ormstack.oc1..examplej6whb4mluogzdv4xjma ",
    "time-created": "2018-03-09T20:52:13.922000+00:00",
    "time-finished": null,
    "variables": {
      "compartment_ocid": "ocid1.compartment.oc1..example7lcdlihrvur7xq",
      "region": "us-phoenix-1"
    }P
  }
}
```

Review the Execution Plan

The content of the plan is viewable in the job's log file. To view log entries for a given job, use the `job get-job-logs` command and provide the job OCID, as shown here:

```
oci resource-manager job get-job-logs --job-id <plan_job_OCID>
```

The command returns JSON objects that describe log entries. Each object has a message member with a property that displays one line of the execution plan. In the example shown below, the plan creates a single virtual cloud network (VCN); the remaining members show details about the VCN.

```
...
 {
   "level": "INFO",
   "message": "Terraform will perform the following actions:",
   "timestamp": "2018-05-24T00:57:14.170000+00:00",
   "type": "TERRAFORM_CONSOLE"
 },
 {
   "level": "INFO",
   "message": "",
   "timestamp": "2018-05-24T00:57:14.170000+00:00",
   "type": "TERRAFORM_CONSOLE"
 },
 {
   "level": "INFO",
   "message": "+ oci_core_virtual_network.vcn1",
   "timestamp": "2018-05-24T00:57:14.170000+00:00",
   "type": "TERRAFORM_CONSOLE"
 },
```

```
{
  "level": "INFO",
  "message": "id: <computed>",
  "timestamp": "2018-05-24T00:57:14.172000+00:00",
  "type": "TERRAFORM_CONSOLE"
},
{
  "level": "INFO",
  "message": "cidr_block:  \"10.0.0.0/16\",
  "timestamp": "2018-05-24T00:57:14.172000+00:00",
  "type": "TERRAFORM_CONSOLE"
},
{
  "level": "INFO",
  "message": "compartment_id:
\"ocid1.tenancy.oc1..exampleaqnpcpfqfmrf6dw5gcew7yqpirvarueirj2mv4jzn5goejsxma\",
  "timestamp": "2018-05-24T00:57:14.172000+00:00",
  "type": "TERRAFORM_CONSOLE"
},
{
  "level": "INFO",
  "message": "default_dhcp_options_id:  <computed_value>",
  "timestamp": "2018-05-24T00:57:14.172000+00:00",
  "type": "TERRAFORM_CONSOLE"
},
{
  "level": "INFO",
  "message": "      default_route_table_id: <computed_value>",
  "timestamp": "2018-05-24T00:57:14.172000+00:00",
  "type": "TERRAFORM_CONSOLE"
},
{
  "level": "INFO",
  "message": "      default_security_list_id: <computed_value>",
  "timestamp": "2018-05-24T00:57:14.172000+00:00",
  "type": "TERRAFORM_CONSOLE"
},
...
```

Updating the Execution Plan

To update the execution plan, follow these steps:

1. Open the Terraform configuration (.tf) file and update the configuration as desired.
2. Update the stack to use the revised configuration file by using `stack update` (`oci resource-manager stack update` *<options>*).
3. Run a new plan job against the stack.
4. Review the resulting execution plan (log files) and confirm the output.

Run an Apply Job

Running an apply job takes the execution plan and runs (applies) it against the stack. To create and run the apply job, call the `job create` command and pass in the plan job OCID, the stack OCID, the operation type, and optionally a display name, as shown here:

```
oci resource-manager job create --operation APPLY
    --apply-job-plan-resolution <JSON string>
    --stack-id <stack_OCID>
    --display-name "Example Apply Job"
```

It is strongly recommended that you run a plan job before you run an apply job, as you will need to provide a plan job OCID as an input parameter on your apply job command, `--apply-job-plan-resolution`. Note that `--apply-job-plan-resolution` is a complex type that takes a JSON string as input. This JSON string can take only one of three available parameters, `isAutoApproved`, `isUseLatestJobId`, or `planJobId`, as shown:

```
{
  "planJobId": "<plan_job_OCID>"
  "isAutoApproved": false,
  "isUseLatestJobId": false,
}
```

Depending on which approach you use, do one of the following:

- Set the `planJobId` parameter to the plan job OCID.
- Set `isAutoApproved` to `true` to use the Terraform configuration directly without reference to an execution plan.
- Set the `isLatestJobId` to `true` to use the most recently run plan job

Following is an example of JSON input when using the `planJobId` parameter:

```
oci resource-manager job create --stack-id <stack_OCID> --apply-job-plan-resolution '
{"planJobId":"<plan_job_OCID>"}' --operation APPLY
```

Following is an example of JSON input when using the `isAutoApproved` parameter:

```
oci resource-manager job create --stack-id <stack_OCID> --apply-job-plan-resolution '
{"isAutoApproved":true}' --operation APPLY
```

Periodically check the lifecycle state of your apply job to see when it switches from `IN_PROGRESS` to `SUCCEEDED`. To do so, use the `job get` command and provide a job OCID, as shown here:

```
oci resource-manager job get --job-id <apply_job_OCID>
```

View State

When you run an apply job, Resource Manager creates a state file, which you can then download and inspect by calling `job get-job-tf-state` and passing in the job OCID, as shown:

```
oci resource-manager job get-job-tf-state --job-id <apply_job_OCID>
```

This call returns a JSON representation of the state file. Following is an example state file:

```
{
  "data": {
    "lineage": "57ef4f0c-c8cd-8a32-d45f-d2c40be7b915",
    "modules": [
      {
        "depends_on": [],
        "outputs": {},
        "path": [
          "root"
        ],
        "resources": {
          "oci_core_virtual_network.vcn1": {
            "depends_on": [],
            "deposed": [],
            "primary": {
              "attributes": {
                "cidr_block": "10.0.0.0/16",
                "compartment_id": "ocid1.tenancy.oc1..examplerueirj2mv4jzn5vkjgoejsxma",
                "default_dhcp_options_id": "ocid1.dhcpoptions.oc1.phx.exampletlwtdnjp5nuixopfwq",
```

```
                "default_route_table_id": "ocid1.routetable.oc1.phx.examplehttsmjqc6vgnq7dw75h6q6oq",
                "default_security_list_id":
"ocid1.securitylist.oc1.phx.exampleztvxuauezzmz4qxsfg6e4c5l7dlq",
                "display_name": "My VCN display name",
                "dns_label": "myvcntest",
                "id": "ocid1.vcn.oc1.phx.examplegjih5mjjkb3g777aza",
                "state": "AVAILABLE",
                "time_created": "2018-05-24 01:13:05.855 +0000 UTC",
                "vcn_domain_name": "myvcntest.oraclevcn.com"
              },
              "id": "ocid1.vcn.oc1.phx. examplegjih5mjjkb3g777aza",
              "meta": {
                "e2bfb730-ecaa-11e6-8f88-34363bc7c4c0": {
                  "create": 300000000000,
                  "delete": 300000000000,
                  "update": 300000000000
                }
              },
              "tainted": false
            },
            "provider": "provider.oci",
            "type": "oci_core_virtual_network"
          }
        }
      }
    ],
    "serial": 4,
    "terraform_version": "0.11.7",
    "version": 3
  }
}
```

Inspect Resources on the Stack

After provisioning your stack with an apply job, you can inspect the cloud resources that you've provisioned by issuing the following command and specifying which resource you wish to inspect. In our example configuration, we are looking at list of VCNs in the compartment to confirm that the one you created is in the list.

```
oci network vcn list
    --compartment-id <vcn_compartment_OCID>
```

Run a Destroy Job

You can tear down and clean up the resources you've created by running a destroy job and specifying the stack OCID, as shown here:

```
oci resource-manager job create --operation DESTROY --stack-id <stack_OCID> --apply-job-plan-resolution
'{"isAutoApproved": true}'
```

To confirm that the cloud resources have been deleted, list the VCNs in the compartment to confirm that your VCN no longer appears in the list:

```
oci network vcn list
    --compartment-id <vcn_compartment_OCID>
```

Using Remote Exec

With Resource Manager, you can use Terraform's remote exec functionality to execute scripts or commands on a remote computer. You can also use this technique for other provisioners that require access to the remote resource.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Prerequisites

- The location where the script is remotely executed must be an Oracle Cloud Infrastructure resource that has a public IP and supports remote login.
- On Windows, WinRM must be enabled. On Linux or Unix, SSH must be enabled.
- A key pair used for signing API requests, with the public key uploaded to Oracle. For more information on generating and uploading keys, see Required Keys and OCIDs.

Authenticating

We recommend using one of the following approaches, depending on whether you have access to the Key Management service. For more information, see Overview of Key Management

With Key Management

First, use Key Management to encrypt your private key. For more information, see Managing Keys and Using Keys.

Next, provide the encrypted private key to Resource Manager. You can use the decrypt data source to decrypt it.

The following code sample demonstrates this process.

```
data "oci_kms_decrypted_data" "private_key_decrypted" {
    #Required
    ciphertext = "${file(var.encrypted_private_key_path)}"
    crypto_endpoint = "${var.decrypted_data_crypto_endpoint}"
    key_id = "${var.kms_encryption_key_id}"
}


resource "oci_core_instance" "TFInstance1" {
  availability_domain = "${lookup(data.oci_identity_availability_domains.ADs.availability_domains
[var.availability_domain - 1],"name")}"
  compartment_id      = "${var.compartment_ocid}"
  display_name        = "TFInstance"
  hostname_label      = "instance3"
  shape               = "${var.instance_shape}"
  subnet_id           = "${oci_core_subnet.ExampleSubnet.id}"

  source_details {
    source_type = "image"
    source_id   = "${var.instance_image_ocid[var.region]}"
  }

  extended_metadata {
    ssh_authorized_keys = "${var.ssh_public_key}"
  }
}
```

```
resource "null_resource" "remote-exec" {
      connection {
      agent       = false
      timeout     = "30m"
      host        = "${oci_core_instance.TFInstance1.public_ip}"
      user        = "${var.opc_user_name}"
      private_key = "${data.oci_kms_decrypted_data.test_decrypted_data.plaintext}"
    }

    inline = [
      "touch ~/IMadeAFile.Right.Here"
    ]

  }
```

Without Key Management

If you do not have access to the Key Management service, you can dynamically generate a key pair and store them in the state file.

1. Generate a key pair using a TLS resource.
2. When you launch the Compute instance, use the public key from the TLS resource.
3. When you establish the SSH connection, provide the private key.



Warning

You should not save your private key in your Terraform configuration file because that is not a secure location.

The following sample demonstrates how to use the TLS private key resource to provision a compute instance, then perform a remote execution on that instance.

```
resource "tls_private_key" "public_private_key_pair" {
  algorithm   = "RSA"
}
```

```
resource "oci_core_instance" "TFInstance1" {
  availability_domain = "${lookup(data.oci_identity_availability_domains.ADs.availability_domains
[var.availability_domain - 1],"name")}"
  compartment_id      = "${var.compartment_ocid}"
  display_name        = "TFInstance"
  hostname_label      = "instance3"
  shape               = "${var.instance_shape}"
  subnet_id           = "${oci_core_subnet.ExampleSubnet.id}"

  source_details {
    source_type = "image"
    source_id   = "${var.instance_image_ocid[var.region]}"
  }

  extended_metadata {
    ssh_authorized_keys = "${tls_private_key.public_private_key_pair.public_key_openssh}"
  }
}

resource "null_resource" "remote-exec" {
  depends_on = ["oci_core_instance.TFInstance1"]

  provisioner "remote-exec" {
    connection {
      agent       = false
      timeout     = "30m"
      host        = "${oci_core_instance.TFInstance1.public_ip}"
      user        = "${var.opc_user_name}"
      private_key = "${tls_private_key.public_private_key_pair.private_key_pem}"
    }

    inline = [
      "touch ~/IMadeAFile.Right.Here"
    ]
  }
}
```

Connection Construct

This example of demonstrates how to use a `connection` construct for remote-exec. Terraform uses a number of defaults when connecting to a resource, but these can be

overridden using a `connection` block in either a `resource` or `provisioner`. For more information, see Provisioner Connections.

Sample: Creating a Compute Instance Using Resource Manager

This sample provides an end-to-end walkthrough of the tasks required to create and deploy an Oracle Cloud Infrastructure Compute instance using the Oracle Resource Manager. For a brief introduction to Resource Manager, see Overview of Resource Manager.



Warning

Avoid entering confidential information when assigning descriptions, tags, or friendly names to your cloud resources through the Oracle Cloud Infrastructure Console, API, or CLI.

Highlights

This walkthrough provides samples that demonstrate how to use Oracle Cloud Infrastructure Resource Manager to create a Compute instance. Resource Manager uses Terraform to provision the resources that you've defined in a Terraform configuration. The resources are organized into stacks, which you create and provision using jobs.

The walkthrough covers four general tasks:

- Create a Terraform configuration using the HashiCorp configuration language (HCL). For more information, see Configuration Syntax.
- Create a stack in which to provision your infrastructure.
- Run a plan job against your stack, which parses your configuration and creates a Terraform execution plan.

- Run an apply job, which uses the execution plan to provision the stack with the resources that you defined in the Terraform configuration.

Before We Begin

Ensure that you have installed, obtained, or created the prerequisites:

- An Oracle Cloud Infrastructure tenancy
- The Oracle cloud ID (OCID) for the compartment where you wish to create your stack.
- A user account that includes the following:
  - An API signing key. For guidance, see Required Keys and OCIDs.
  - Required Identity and Access Management (IAM) permissions. For more information, see How Policies Work and Details for Resource Manager.
- If you wish to use the Oracle Cloud Infrastructure CLI instead of the Console, you will need to install and configure it. For details on setting up the CLI, see the Quickstart and Configuration

Create the Terraform Configuration



Important

This section applies whether you are using either the Console or the command line interface (CLI).

A Terraform configuration is a set of files that define your Terraform provider, specify the resources you intend to provision, declare variables, and provide specific instructions for provisioning the resources. The configuration files are bundled into a .zip file and uploaded to Resource Manager. For more information, see Terraform Configuration.

In this example, our configuration uses several configurations (.tf) files to direct Resource Manager to execute the following sequence of operations:

Create an Oracle Cloud Infrastructure Provider

The following code sample creates a basic Oracle Cloud Infrastructure Terraform provider. You can provide values as variables that are defined either in a variables file, or in the provider definition (.tf) file. For more information, see Provider Configuration.

```
provider "oci" {
  region = "${var.region}"
}
```

Define Variables

You must define the variables that you use in the configuration files. A best practice is to create a "variables" file in the configuration package that you upload. Following is an example from a configuration file that we've named "variables.tf". For more information about using variables, see Input Variables. See also Configuring Input Variables.



Warning

Do not include confidential information in variables in the configuration files.

```
variable "compartment_ocid" {
       default = "ocid1.compartment.oc1..examplean6gzrs3icfftea5nuqrc2znxkbpzet7t64nqhu5mbrewhhrgc73a"
}

variable "region" {
       default = "us-phoenix-1"
}

variable "InstanceImageOCID" {
    type = "map"
    default = {
        // See https://docs.cloud.oracle.com/images/
        // Oracle-provided image "Oracle-Linux-7.5-2018.10.16-0"
        us-phoenix-1 =
"ocid1.image.oc1.phx.aaaaaaaaoqj42sokaoh42l76wsyhn3k2beuntrh5maj3gmgmzeyr55zzrwwa"
```

```
        us-ashburn-1 =
"ocid1.image.oc1.iad.aaaaaaaageeenzyuxgia726xur4ztaoxbxyjlxogdhreu3ngfj2gji3bayda"
        eu-frankfurt-1 = "ocid1.image.oc1.eu-frankfurt-
1.aaaaaaaaitzn6tdyjer7jl34h2ujz74jwy5nkbukbh55ekp6oyzwrtfa4zma"
        uk-london-1 = "ocid1.image.oc1.uk-london-
1.aaaaaaaa32voyikkkzfxyo4xbdmadc2dmvorfxxgdhpnk6dw64fa3l4jh7wa"
    }
}

variable "ssh_public_key" {
       default = "ssh-rsa <public_key_value>"
}

# Defines the number of instances to deploy
variable "NumInstances" {
    default = "1"
}

variable "InstanceShape" {
    default = "VM.Standard2.1"
}

# Specifies the Availability Domain
variable "localAD" {
    default = "<AD_name>"
}
```

For more information about variables declared in the preceding examples, see the following:

- `InstanceImageOCID`: Oracle-Provided Images
- `InstanceShape`: Compute Shapes
- `region` and `localAD`: Regions and Availability Domains

Create a Virtual Cloud Network (VCN)

The following code sample creates an Oracle Cloud Infrastructure virtual cloud network (VCN) named "ExampleVCN."

```
resource "oci_core_virtual_network" "ExampleVCN" {
  cidr_block = "10.1.0.0/16"
  compartment_id = "${var.compartment_ocid}"
  display_name = "TFExampleVCN"
  dns_label = "tfexamplevcn"
}
```

Create a Subnet in Your VCN

The following code sample creates a subnet named "ExampleSubnet" in VCN that we just created.

```
resource "oci_core_subnet" "ExampleSubnet" {
  availability_domain = "${var.localAD}"
  cidr_block = "10.1.20.0/24"
  display_name = "TFExampleSubnet"
  dns_label = "tfexamplesubnet"
  security_list_ids = ["${oci_core_virtual_network.ExampleVCN.default_security_list_id}"]
  compartment_id = "${var.compartment_ocid}"
  vcn_id = "${oci_core_virtual_network.ExampleVCN.id}"
  route_table_id = "${oci_core_route_table.ExampleRT.id}"
  dhcp_options_id = "${oci_core_virtual_network.ExampleVCN.default_dhcp_options_id}"
}
```

Create an Internet Gateway

The following code sample creates an internet gateway named "ExampleIG" in the VCN that we created.

```
resource "oci_core_internet_gateway" "ExampleIG" {
  compartment_id = "${var.compartment_ocid}"
  display_name = "TFExampleIG"
  vcn_id = "${oci_core_virtual_network.ExampleVCN.id}"
}
```

Create a Core Route Table

The following code sample creates a Oracle Cloud Infrastructure core route table in the VCN and then applies two route rules.

```
resource "oci_core_route_table" "ExampleRT" {
  compartment_id = "${var.compartment_ocid}"
  vcn_id = "${oci_core_virtual_network.ExampleVCN.id}"
  display_name = "TFExampleRouteTable"
  route_rules {
    cidr_block = "0.0.0.0/0"
```

```
    network_entity_id = "${oci_core_internet_gateway.ExampleIG.id}"
  }
}
```

Create a Compute Instance

The following extended code example creates an Oracle Cloud Infrastructure Compute instance. The code also references the image on which the Compute instance is created, sets boot volume size, adds essential metadata, and applies both free-form and defined tags.

```
resource "oci_core_instance" "TFInstance" {
  count = "${var.NumInstances}"
  availability_domain = "${var.localAD}"
  compartment_id = "${var.compartment_ocid}"
  display_name = "TFInstance${count.index}"
  shape = "${var.InstanceShape}"

  create_vnic_details {
    subnet_id = "${oci_core_subnet.ExampleSubnet.id}"
    display_name = "primaryvnic"
    assign_public_ip = true
    hostname_label = "tfexampleinstance${count.index}"
  },

  source_details {
    source_type = "image"
    source_id = "${var.InstanceImageOCID[var.region]}"

    # Apply this to set the size of the boot volume that's created for this instance.
    # Otherwise, the default boot volume size of the image is used.
    # This should only be specified when source_type is set to "image".
    #boot_volume_size_in_gbs = "60"
  }

  # Apply the following flag only if you wish to preserve the attached boot volume upon destroying this
instance
  # Setting this and destroying the instance will result in a boot volume that should be managed outside
of this config.
  # When changing this value, make sure to run 'terraform apply' so that it takes effect before the
resource is destroyed.
```

```
  #preserve_boot_volume = true

  metadata {
    ssh_authorized_keys = "${var.ssh_public_key}"

  }
  timeouts {
    create = "60m"
  }
}
```

Finalize the Configuration

Ensure that all of the configuration files are in a single directory. When using the CLI, you create the configuration .zip file, then specify it using the `--config-source` parameter. When using the Console, you create the configuration .zip file , then upload it.



Important

When you create the configuration .zip file, you must ensure that all of the configuration (*.tf) files are at the root level. Resource Manager can't locate configuration files that are in directories inside the .zip file.

Build, Deploy, and Provision the Infrastructure

Use your Terraform configuration to build and deploy your infrastructure by taking the following actions:

1. Create a stack in a tenancy compartment of your choosing.
2. Run a plan job against the configuration. This creates and "execution plan," which is a step-by-step representation of the deployment in job log entries.

3. Review the execution plan to confirm that it represents your intentions. If changes are needed, revise the configuration, then rerun the plan job to obtain an updated execution plan.
4. Run an apply job against the execution plan. This provisions the stack that you created with the resources that you specified in your configuration.
5. You can review the job's log entries. You also have access to Terraform state, which represents the current state of the deployment.

Create a Stack

Stacks are collections of resources that you can act on as a group. All of the resources that you specify in your configuration are provisioned in the stack that you create.

Create a Stack Using the Console

1. Open the navigation menu. Under **Resource Manager**, select **Stacks**.
2. In the **Compartment** drop-down, ensure that you have the correct compartment selected.
3. Click **Create Stack**. This opens the **Create Stack** form.
    a. Confirm that you are in the correct compartment.
    b. Provide a name for the stack.
    c. Add an optional description.
    d. Upload your configuration .zip file by dropping the file on the form, or by navigating to the file in your file system.
    e. Create variables, as needed.
4. Click **Create**.



Note

You can return later to update stack settings or add variables after you have created the stack.

Create a Stack Using the CLI

To create a stack, execute the `stack create` command. Specify the compartment in which to create the stack by providing the compartment OCID, and provide a path to the .zip file that contains your configuration.

```
oci resource-manager stack create --compartment-id <compartment_OCID>
    --config-source <path_to_config_zip_file>
    --display-name "Sample Configuration"
    --description "Example: Create a Compute Instance"
    --working-directory: ""
```

To specify the working directory in the .zip file, use the optional parameter `--working-directory` (which is null in the code sample). When not specified, the root directory within the .zip file is considered to be the working directory.

The response to your call to `stack create` includes the OCID of your new stack, along with other important metadata. Following is an example response:

```
{
  "data": {
    "config-source": {
      "working-directory": null,
      "config-source-type": "ZIP_UPLOAD"
    },
    "defined-tags": {},
    "description": "Create a Compute instance",
    "display-name": "Sample Configuration",
    "freeform-tags": {},
    "id": "ocid1.ormstack.oc1..examplej6whb4mluogzdv4xjma",
    "lifecycle-state": "ACTIVE",
    "time-created": "2018-08-03T18:26:56.299000+00:00",
```

```
    "variables": {
      "compartment_ocid": "ocid1.compartment.oc1..example7lcdlihrvur7xq",
      "region": "us-phoenix-1"
    }
  }
}
```

Create, Edit, or Delete Variables Using the Console

In the Console, variables are assigned by creating key/value pairs when you create a stack, or when you edit an existing stack.

Create a Variable When Creating a New Stack

1. In Resource Manager, click **Create Stack** and provide required information.
2. In the **Variables** text box, click **+Additional Variable** to add a variable.
3. Enter the variable **Key** and **Value** in the respective text boxes.
4. To create another variable, click **+Additional Variable**. Otherwise, click **Create**.

Create, Edit, or Delete a Variable When Editing an Existing Stack

1. Select the stack for which you wish to create, edit, or delete a variable by clicking the stack's link on the **Stacks** page. This opens the **Stack Details** page for the stack.
2. Under **Resources** on the left navigation area, click **Variables**. This displays a list of variables currently defined for the stack.
3. Click **Edit Variables**. This opens the **Edit Variables** form.
4. To edit a variable, modify the text in the **Key** or **Value** text box, as needed.
5. To delete a variable, click the red box at the right of the variable value.
6. To create a variable, click **+Additional Variable** and enter the appropriate **Key** and **Value**.
7. Click **Save**.

Run a Plan Job

The plan job parses your configuration and creates an execution plan, but does not act on the stack. Once the plan job has completed, you can evaluate the execution plan by viewing the job's log entries to confirm that it performs the expected operations, and in the intended sequence.



Important

To update the execution plan after running the plan job, you must first update the configuration and recreate the configuration .zip file. Then, upload the new .zip file and rerun the plan job.

Run a Plan Job Using the Console

1. Open the navigation menu. Under **Resource Manager**, select **Stacks**. This opens a list of stacks in the compartment.
2. Click on the stack name (a hyperlink), which opens the **Stack Details** page for the specified stack.
3. From the **Terraform Actions** drop-down, select **Plan**.
4. In the form that opens, enter a name for the plan job, then click **Plan**.

This action creates and runs the plan job and lists it on the **Jobs** page. The list entry indicates the job type, its current state, shows the job OCID, and lists the job start and end times.



Note

Clicking the **Actions** icon (three vertical dots) on the right of each job list item enables downloading the logs, the Terraform state file, and the execution plan. The execution plan is a binary file, and therefore not viewable. However, the log file provides a representation of the execution plan.

Run a Plan Job Using the CLI

Once your stack is created, run a plan job against the stack. The plan job consumes your Terraform configuration and creates an execution plan.

To run a plan job, call `job create`, specify the operation type (`PLAN`), and pass the stack OCID, as shown in the example:

```
oci resource-manager job create --operation PLAN
  --stack-id <stack_OCID>
  --display-name "Example Compute Instance"
```

Depending on the complexity of the configuration, the plan job can take several minutes to complete. To check the operation's status, you can call `job get` and pass in the job OCID, which was returned in the response to the preceding call to `job create`:

```
oci resource-manager job get --job-id <plan_job_OCID>
```

The call returns one of the following `lifecycleState` values:

- `ACCEPTED`: The job is queued for execution.
- `IN_PROGRESS`: The job is running.
- `FAILED`: The job has failed and stopped running.
- `SUCCEEDED`: The job has completed successfully.

- `CANCELING`: The job has been notified to cancel, but has not yet stopped running.
- `CANCELED`: The job was canceled and has stopped running.

Review the execution plan to ensure that it accurately reflects your intentions. You can review it indirectly by calling `get-job-logs` and reviewing the "message" field in the sequence of log entries. These values represent the sequence of operations specified in your configuration.

```
oci resource-manager job get-job-logs --job-id <plan_job_OCID>
```

If you see problems or errors and wish to make changes, do the following:

1. Update the appropriate configuration (.tf) file.
2. Direct the stack to use the revised configuration by calling the `stack update` command.
3. Rerun your plan job against the updated stack.
4. When the new plan job completes, call `get-job-logs` again to return the revised execution plan (log entries).

Run the Apply Job

When satisfied with the execution plan, we're ready to do the work of provisioning the stack with the resources that we've defined. The apply job takes the execution plan and "applies" it to the stack. The result is a fully provisioned stack.

Run the Apply Job Using the Console

1. Open the navigation menu. Under **Resource Manager**, select **Stacks**. This opens a list of stacks in the compartment.
2. Click on the desired stack name (a hyperlink), which opens the **Stack Details** page for the specified stack.
3. From the **Terraform Actions** drop-down, select **Apply**. You're presented with a confirmation dialog. Click **Apply**.
4. Monitor the status of the job in the **Jobs** list on the **Stack Details** page. The job has completed when the **State** column shows "Succeeded."

5. While the job runs, or after it completes, you can click on the job name (a hyperlink) to enter the **Job Details** page, where you can review the job log. You can download the log file by clicking **Download Logs**.

Run the Apply Job Using the CLI

Once you have confirmed the execution plan, run an apply job. This "applies" the execution plan to the stack, executes the instructions, and provisions the stack with the specified resources. To create and run the apply job, call `job create`, specify the operation type (`APPLY`), the `apply-job-plan-resolution` value, then provide both the plan job and stack OCIDs. You can optionally provide a display name:

```
oci resource-manager job create
  --operation APPLY
  --apply-job-plan-resolution '{ "planJobId": "<plan_job_OCID>"}'
  --stack-id <stack_OCID>
  --display-name "Sample Apply Job"
```

Depending on the complexity of your execution plan, the operation can take some time. You can check the status of the job by calling `job get` and providing the job OCID, which was provided in the response to the `job create` call. This call returns one of six `lifecycleState` values described earlier.

```
oci resource-manager job get --job-id <apply_job_OCID>
```

Review State

Essential information about the state of your resources configuration is maintained in a state file, which is a JSON file. The state file maps your stack's resources to your configuration and also maintains essential configuration metadata, such as resource dependencies. Resource Manager generates and updates state files automatically.

The Resource Manager supports state locking by allowing only one job at a time to run on a given stack. For more information about state files, see Hashicorp: State.

Review State Using the Console

You can get the current state of job on its Job Details page. The Job Information tab displays the current state of the job. Click the Download Terraform State button to download the job's state (.json) file.

Review State Using the CLI

When you run an apply job, Resource Manager creates a "state" file, which is a representation of the job's output in JSON format. You can obtain the state file by calling `get-job-tf-state` and providing the job OCID:

```
oci resource-manager job get-job-tf-state --job-id <apply_job_OCID>
```

About Destroy Jobs

If necessary, you can tear down the stack you created and clean up the resource that it contains by running a destroy job.

Run a Destroy Job Using the Console

1. Open the navigation menu. Under **Resource Manager**, select **Stacks**. This opens a list of stacks in the compartment.
2. Click the desired stack name (a hyperlink), which opens the **Stack Details** page for the specified stack.
3. From the **Terraform Actions** drop-down, select **Destroy**. You're presented with a confirmation dialog. Click **Destroy**.

Run a Destroy Job Using the CLI

Run a destroy job using the CLI by calling `job create` and specifying the `DESTROY` operation type. Provide the stack OCID and set the `--apply-job-plan-resolution` parameter to `'{"isAutoApproved: true}'`:

```
oci resource-manager job create
  --operation DESTROY
  --stack-id <stack_OCID>
  --apply-job-plan-resolution '{"isAutoApproved: true}'
```

CHAPTER 23 Search

This chapter explains how to search for resources across compartments.

Overview of Search

Oracle Cloud Infrastructure Search lets you find resources in your tenancy without requiring you to navigate through different services and compartments. You do not need to know the compartment or availability domain where a resource exists in order to locate and view its details. Rather, with a query, you can use as little as a single piece of information, such as the creation date or other supported attribute, to find a resource. Querying also helps you avoid the latency associated with loading a long list of results onto a single page or the inconvenience of viewing a long list that spans multiple pages.

You might find it helpful to use Search to find related resources when creating or deleting another resource. For example, you might want to find what compartments already exist before creating a new one because compartments cannot be deleted. Or, if you want to delete a volume, you can use a query to verify that a backup exists.

Another benefit of Search is that you can find resources that require action. For example, you might want to delete terminated block volumes because you no longer need them and don’t want them to count against your service limits. Or, you can search for all resources that match a specific naming scheme, in case you want to act on a category of associated resources. Sometimes, resources in a specific lifecycle state, such as databases in a failed state, require troubleshooting. With Search, you can quickly identify those resources and resolve problems.

Supported Resources

Search supports queries for the following Oracle Cloud Infrastructure services and resources. This table will be updated as query support is added for more resources.

Service	Resource Type	Attributes
Block Volume	`bootvolume`	See BootVolume Reference.
Block Volume	`bootvolumebackup`	See BootVolumeBackup Reference.
Block Volume	`volume`	See Volume Reference.
Block Volume	`volumebackup`	See VolumeBackup Reference.
Compute	`image`	See Image Reference.
Compute	`instance`	See Instance Reference.
Compute	`consolehistory`	See ConsoleHistory Reference.
Database	`autonomousdatabase`	See AutonomousDatabase Reference.
Database	`database`	See Database Reference.
Database	`dbsystem`	See DbSystem Reference.
IAM	`compartment`	See Compartment Reference.
IAM	`group`	See Group Reference.
IAM	`identityprovider`	See IdentityProvider Reference.
IAM	`user`	See User Reference.
Monitoring	`alarm`	See Search-Supported Attributes for Alarms.
Networking	`routetable`	See RouteTable Reference.
Networking	`securitylist`	See SecurityList Reference.
Networking	`subnet`	See Subnet Reference.
Networking	`vcn`	See Vcn Reference.

Service	Resource Type	Attributes
Object Storage	`bucket`	See Bucket Reference.
Resource Manager	`job`	See Job Reference.
Resource Manager	`stack`	See Stack Reference.

Although you can use the query language to search fields and values for any supported attribute, query results only provide information about the following resource attributes:

- Resource type
- Oracle Cloud Identifier (OCID)
- Compartment
- Availability domain
- Display name
- Creation date and time
- Lifecycle state
- Tags (visible in the API only)

The preceding attributes are common to most Oracle Cloud Infrastructure resources. Their meaning is consistent across resource types. Query results do not contain information specific to any resource type. For example, you can query for volumes of a certain size, but search results will not display the **Size** attribute. You must view the details of a resource to see resource-specific information.



Tip

If you use the Console, neither query results nor resource details will include either defined tags or free-form tags, due to display constraints. Any given resource might contain hundreds of tags. If you want to see tags, use the API to view resource details.

Required IAM Permissions

The resources that you see in query results depend on the permissions you have in place for the resource type. You do not necessarily see results for everything in the compartment or tenancy. For example, if your user account is not associated with a policy that grants you the ability to, at a minimum, `inspect` the `dbsystem` resource type, then you can't query for DB systems. (The verb `inspect` lets you list and get resources.) Instead, Search will show no results for queries of DB system resources. For more information about policies, see How Policies Work. For information about the specific permissions required for the list API operation for your desired resource type, see the Policy Reference for the appropriate service.

Search Language Syntax

This topic describes the basics of the query language for Search, including an explanation of syntax and rules so you can create your own queries. Queries apply search conditions to specific resource types and let you sort results. If you want to search across all supported resource types and resource attributes and do not need ordered search results, you do not need to construct a query. Instead, you can search for a partial or exact match of free-form text without applying query language syntax to your search.

When you are ready to run a query, see Querying Resources for instructions.

Query Basics

The following examples show the basic syntax of a query:

```
query <resourceType> resources where <conditions> sorted by <fieldName> <order>
```

Or:

```
query <resourceType> resources matching <keywords>
```

Search ignores white space, indentation, and line breaks. Sample queries include indentation to improve readability. For the purposes of demonstrating syntax only, angle brackets (<>) and italicized text indicate variables, which can consist of one or more keywords.

In a query, clauses include the following:

- `query` - (Required) Selects which resources to return based on subsequent clauses. Query statements always begin with the word `query`.
- `where` - Matches resources to the specified `conditions`.
- `matching` - Matches resources to the specified text regardless of whether the text matches exactly, matches the resource type, or appears in an indexed resource attribute.
- `sorted by` - Orders resources according to `fieldName` in the order specified by `order`. If you do not include this clause, Search lists results by creation date in descending order, with the newest resources listed first.

Clauses are optional unless indicated otherwise. For matching purposes, you can use the `where` clause and the `matching` clause either separately or together.

In the `query` clause, you specify the following information:

- `resourceType` - (Required) Specifies the resource type to which the subsequent clauses apply when you run the query. You can specify either the resource type name (for example, `database` or `group`) or `all`. If you specify `all`, Search searches for the `conditions` against all resource types. You can query for individual resource types, but not family types. For a list of supported resource types, see the **Supported Resources** section of Overview of Search.
- `resources` - (Required) Specifies that this is a resource query.

Conditions

The `where` clause applies `conditions` that act as a filter to restrict the results returned by Search. You can specify one or more condition statements. For more information about multiple conditions, see Grouping Conditions.

In a query, `conditions` consist of the following:

```
<fieldName> <operation> <value>
```

The `fieldName` keyword is the resource attribute against which the `operation` and desired `value` of that attribute are evaluated. Each field is associated with a field type. The type of `operation` you can use in your `conditions` filter depends on the field type. The API documentation includes a reference for each supported resource type that specifies attributes, their field types, and any restrictions. For more information, see the **Supported Resources** section of Overview of Resource Query.

In query `conditions`, an `operation` is a comparison operator that applies to the `value` in the statement. The `value` keyword refers to the value of the `fieldName` you specified. Search evaluates whether the specified attribute of the desired resource type matches or does not match the desired `value`, according to the operation. You must enclose a value in opening and closing straight single quotes (').

The following table describes supported operations for resource queries:

Operation	Description	Supported Field Types	Case-sensitive?	Example
=	Equals, or exact matching for strings	String, integer, rational, Boolean, date-time	No	If the `value` was '**backUp**', it would match "backup", "BACKUP", "BackUp", "backUp", or any other variation in casing.
!=	Does not equal	String, integer, rational, Boolean, date-time	No	If the `value` was '**backUp**', it would match anything that does not equal "backUp", "backup", or any other variation in casing. It also would match anything that does not contain the characters 'backup' in that order.
==	Strictly equals	String	Yes	If the value was '**backUp**', it would only match "backUp" and no other variation in casing.
!==	Strictly does not equal	String	Yes	If the `value` was '**backUp**', it would match "backup", "BACKup", or anything except "backUp", with that exact casing.
>=	Greater than or equal to	Integer, rational, date-time	Not applicable	For a query where you have `size >= 5` as the condition, all results have a value of **5** or greater in the field named **size**.

Operation	Description	Supported Field Types	Case-sensitive?	Example
>	Greater than	Integer, rational, date-time	Not applicable	For a query where you have `size > 5` as the condition, all results have a value of greater than **5** in the field named **size**.
<=	Less than or equal to	Integer, rational, date-time	Not applicable	For a query where you have `size <= 5` as the condition, all results have a value of **5** or less in the field named **size**.
<	Less than	Integer, rational, date-time	Not applicable	For a query where you have `size < 5` as the condition, all results have a value of **5** or less in the field named **size**.

Date and Time Values

You can specify date and time values by using any of the following pattern string formats:

Format	Examples	Comments
<yyyy>-*<MM>*-*<dd>* *<HH>*:*<mm>*:*<ss>* *<TimeZone>*	'2018-06-19 16:15:41 PDT', '2018-06-19 16:15:41 -08:00'	`TimeZone` is optional. If `TimeZone` is omitted, UTC is used.
<EEE>, *<d>* *<MMM>* *<yyyy>* *<HH>*:*<mm>*:*<ss>* *<TimeZone>*	'Tue, 19 Jun 2018 16:15:41 +0300', '19 June 2018 16:15:41'	`TimeZone` is optional. If `TimeZone` is omitted, UTC is used.
<yyyy>-*<MM>*-*<dd>*`T`*<HH>*:*<mm>*:*<ss>*`Z`	2018-06-19T16:15:41Z	Time in UTC. '`T`' and '`Z`' are case-sensitive.

You must observe spacing. Interpret dashes, colons, commas, and the characters 'T' and 'Z' literally. To interpret placeholder values in the preceding table, you can refer to the following pattern syntax:

Letter	Date or Time Component	Presentation
`y`	Year	Year
`M`	Month in year	Month
`d`	Day in month	Day
`H`	Hour in day (from 00-23)	Number
`m`	Minute in hour	Number
`s`	Seconds in minute	Number
`E`	Day in week	Text

Repeating pattern letters indicate their exact presentation. For example, 'HH' means you must use '00' and not '0' to represent midnight. Similarly, 'EEE' means 'Tue' and not 'Tuesday'. Likewise, 'MM' requires '09' instead of '9' to represent the month of September.

`TimeZone` is optional, but in your chosen format, you can specify `TimeZone` in any of the following ways:

- **Name**. You can specify a time zone by its name, such as **GMT** or **PDT**. Values are case-insensitive.
- **GMT offset value**. You can specify a time zone according to its GMT offset. For example, **GMT-08:00**. Values are case-insensitive.
- **ISO 8601 time zone**. You can specify a time zone according to ISO 8601 standards. For example, **-08**, **-0800**, or **-08:00**.

Instead of using one of the preceding formats, you can also specify a date-time value as the constant `now`. The constant `now` represents the current time to the level of granularity of seconds in a minute.

Lastly, you can add or subtract time intervals from any date-time values. For example, you can query for resources that were created within five minutes of a specific time. Search supports the following time intervals:

Letter	Date or Time Component
`s`	Seconds
`m`	Minutes
`h`	Hours
`d`	Days
`w`	Weeks

To specify a time interval in relation to a date-time value, use one of the following formats:

- **now - 3h**
- **2018-06-19 16:15:41 PDT + 1h**

Matching

For matching purposes, instead of or in addition to using a `where` clause with `conditions`, you might want to use the `matching` clause. The `matching` clause obviates the need to specify `conditions` (that contain a field name, operation, and value). A `matching` clause effectively queries all indexed fields by applying the = (equals) operator along with the text you specify. However, it does so without strictly requiring an exact match. For example, the following query uses a `matching` clause to behave the same way as a free text search: `query all resources matching 'instance'`. The query produces results that match all resources and resource attributes that contain the word "instance".

Sorting

The last clause of a resource query is the `sorted by` clause and is optional. The `sorted by` clause orders the results returned by Search based on the field name and lists them according

to the `order` you specify. By default, if you do not specify sort order, results are always sorted by date-time created in descending order.

In the `sorted by` clause, you can specify the following:

- `fieldName` - The field that Search uses to sort results. You can specify any field of any resource. Resources that do not contain the field you specify are listed after the resources that do.
- `order` - You can specify either **asc** or **desc**. Specifying **asc** lists results in ascending order. Specifying **desc** lists results in descending order.

Grouping Conditions

You can group multiple conditions by using either the logical operators `&&` (ampersands, to indicate a logical AND) or `||` (vertical bars, to indicate a logical OR). For example:

```
licenseModel = 'LICENSE_INCLUDED' && dataStoragePercentage > 40 && lifecycleState != 'FAILED'
```

You cannot combine two different logical operators in the same query unless you wrap parentheses around one group of predicates. (Multiple conditions can only use the same logical operator otherwise.) For example:

```
(licenseModel = 'LICENSE_INCLUDED' && dataStoragePercentage > 40) || lifecycleState != 'FAILED'
```

In the preceding example, all results returned will have either “LICENSE_INCLUDED” as the value in the field named “licenseModel” and a value greater than 40 for the field named “dataStoragePercentage” or the value of their “lifecycleState” field name is anything other than “FAILED”.

The following group is also acceptable:

```
licenseModel = 'LICENSE_INCLUDED' && (dataStoragePercentage > 40 || lifecycleState != 'FAILED')
```

In the preceding example, all results returned will have “LICENSE_INCLUDED” as the value in the field named “licenseModel” and either a value greater than 40 as the value for the field named “dataStoragePercentage” or anything that is not “FAILED” for the value of the field named “lifecycleState”.

Search does not perform left-to-right evaluation to reduce ambiguity or clarify intent.

Querying Multiple Resource Types

You can query multiple resource types at once by joining queries. Each query can have its own conditional clause. If the queries that you want to join have different "where" conditions, then the syntax is different from when you have queries for multiple resource types that share the same "where" condition.

The basic syntax for a query for multiple resource types is as follows:

```
query <resourceType>, <resourceType> resources
```

For example:

```
query group, user resources
```

The preceding example query returns all groups and all users in the tenancy.

The following shows the syntax for a query for multiple resource types with conditions, but where the conditions are the same for all resource types:

```
query <resourceType>, <resourceType> resources where <conditions>
```

For example:

```
query group, user resources where displayName = 'administrator'
```

The preceding example query returns all groups with the display name "administrator" and all users with the display name "administrator," with any variation in casing.

If you need to apply differing conditions to any resource type, you must use a `union` keyword instead of comma separation between the joined queries. The following shows the syntax for a query for multiple resource types where some of the resource types share conditions while others do not:

```
query <resourceType>, <resourceType> resources where <conditions> union <resourceType> resources
```

For example:

```
query group, user resources where displayName = 'administrator' union compartment resources
```

The preceding example returns all groups with the display name "administrator" and all users with the display name "administrator," with any variation in casing, and all compartment resources.

Or, for example:

```
query group resources union user resources where displayName = 'administrator' union compartment
resources
```

The preceding example returns all groups and all compartments. It also returns all users with the display name "administrator," with any variation in casing.

Optionally, you can add the `sorted by` clause to the end of the query to order all results in ascending or descending order.

Sample Queries

This topic provides an explanation of sample queries, including what results to expect from a given sample query. For more information about the syntax for constructing a query, see Search Language Syntax.

Example Values

Sample queries show example values for resource attributes. Replace those examples with values from your own tenancy.

Search provides the following sample queries in the Console:

- Query for everything
- Query for everything, sorted by time created
- Query for volumes and users
- Query for volumes and users, sorted by time created
- Query for volumes and users that have any indexed field matching “production," sorted by time created
- Query for all resources that have a specific freeform tag
- Query for all resources that have one of two specific defined tags
- Query for instances in a "Running" state
- Query for instances in either a "Terminated" or "Terminating" state

- Query for all resources in a specific compartment
- Query for all instances due for a maintenance reboot

Query All Resources

Query name: Query for everything

Expected results: Returns all supported resources in the tenancy across all compartments. Lists results, by default, in order of time created, from newest to oldest.

Sample query language:

```
query
 all resources
```

Query All Resources, Sort by Time Created

Query name: Query for everything, sorted by timeCreated

Expected results: Returns all supported resources in the tenancy across all compartments, listed in order of time created, from newest to oldest.

Sample query language:

```
query
 all resources
sorted by timeCreated desc
```

Query Volumes and Users

Query name: Query for volumes and users

Expected results: Returns all block volumes and users in the tenancy. Lists results, by default, in order of time created, from newest to oldest.

Sample query language:

```
query
 volume, user resources
```

Query All Volumes and Users, Sort by Time Created

Query name: Query for volumes and users, sorted by timeCreated

Expected results: Returns all block volumes and users in the tenancy, listed in order of time created, from newest to oldest

Sample query language:

```
query
 volume, user resources
sorted by timeCreated desc
```

Query Volumes and Users Matching "Production," Sorted by Time Created

Query name: Query for volumes and users, with anything matching production, sorted by timeCreated

Expected results: Returns all block volumes and users in the tenancy that have any indexed fields that exactly or partially match the search string "production", irrespective of casing. Lists results, by default, in order of time created, from newest to oldest.

Sample query language:

```
query
 volume, user resources
matching 'production'
sorted by timeCreated desc
```

Query All Resources With Specific Freeform Tags

Query name: Query for all resources that have specific freeform tags

Expected results: Returns all resources in the tenancy that have a freeform tag of "costcenter" with a value of "1234."

Sample query language:

```
query
 all resources
```

```
 where
  (freeformTags.key = 'costcenter' && freeformTags.value = '1234')
```

Query All Resources According to Defined Tags

Query name: Query for all resources that have one of two specific defined tags

Expected results: Returns all resources in the tenancy that have either a tag with the key “region” and value “phx” in the tag namespace “categorization," or all resources in the tenancy that have a tag with the key “region” and value “iad” in the namespace “categorization.” Ignores casing for all keys and values.

Sample query language:

```
query
 all resources
 where
 (definedTags.namespace = 'categorization' && definedTags.key = 'region' && definedTags.value = 'phx')
||
 (definedTags.namespace = 'categorization' && definedTags.key = 'region' && definedTags.value = 'iad')
```

Query Instances According to Specific Lifecycle State

Query name: Query for running instances

Expected results: Returns all instances in the tenancy in a "Running" state. Lists results, by default, in order of time created, from newest to oldest.

Sample query language:

```
query
 instance resources
  where lifeCycleState = 'RUNNING'
```

Query Instances According to One of Two Lifecycle States

Query name: Query for instances terminated or terminating

Expected results: Returns all instances in the tenancy in either a "Terminated" or "Terminating" state. Lists results, by default, in order of time created, from newest to oldest.

Sample query language:

```
query
 instance resources
  where lifeCycleState = 'TERMINATED' || lifeCycleState = 'TERMINATING'
```

Query All Resources According to Compartment ID

Query name: Query for all resources in a compartment

Expected results: Returns all resources in the tenancy with a specific compartment ID. Lists results, by default, in order of time created, from newest to oldest.

Sample query language:

```
query
 all resources
  where compartmentId = 'compartmentOcid'
```

Query All Instances Due for Maintenance Reboot

Query name: Query for all instances which have an upcoming scheduled maintenance reboot

Expected results: Returns all instances in the tenancy with a scheduled maintenance reboot time value of "now." Lists results, by default, in order of time created, from newest to oldest.

Sample query language:

```
query
 instance resources
  where timeMaintenanceRebootDue = 'now'
```

Querying Resources

This topic describes how to find Oracle Cloud Infrastructure resources in your tenancy by performing a free text search or running a query. Queries let you find resources according to

specific fields and conditions while free text searches locate resources with the desired text anywhere in the resource metadata.



Note

Supported Resources and Query Language Syntax

Search bases search results on supported resources. To see what Oracle Cloud Infrastructure services and resources Search supports, see the **Supported Resources** section of Overview of Search.

Furthermore, if free text search results do not produce the results you want, you might need to run a query using query language syntax. For more information about syntax for queries, see Search Language Syntax.

Using the Console

You can find resources by doing one of the following:

- typing free-form text for a free text search
- typing a query (based on resource query language syntax)
- modifying a sample query

By default, text entered into the **Search** box is interpreted as a free text search.

To perform a free text search

1. Open the Console, and then, in the top navigation bar, click **Search**.
2. Type the free-form text you want to search for, and then press ENTER.
3. In the left-hand navigation bar, under **Resource Type**, click the type of supported

resource to filter query results.

4. (Optional) If you do not see the results that you expect, you can refine your search with a query by clicking **Advanced Search**. Then, follow the instructions in To run a custom, free-form query or To run a sample query.

Except for volume backups, compartments, and subnets, you can click the display name of the resource to view details. Search results are eventually consistent, but might not immediately include resources that you created very recently.

To run a custom, free-form query

1. In the Console, if you are not already there, append "/a/query" to the end of your base Console URL. For example, https://console.us-ashburn-1.oraclecloud.com/a/query.
2. In the query text box, type a query using query language syntax, and then click **Search**.
3. In the left-hand navigation bar, under **Resource Type**, click the type of supported resource to filter query results.

Except for volume backups, compartments, and subnets, you can click the display name of the resource to view details. Query results are eventually consistent, but might not immediately include resources that you created very recently.

To run a sample query

1. In the Console, if you are not already there, append "/a/query" to the end of your base Console URL. For example, https://console.us-ashburn-1.oraclecloud.com/a/query.
2. Click **Select Sample Query**, and then click one of the listed sample queries. For an explanation of sample queries, see Sample Queries.
3. Verify that the query language in the query text box satisfies your needs. Change all example values. Add, delete, or modify clauses, as appropriate, and then click **Search**.
4. In the left-hand navigation bar, under **Resource Type**, click the type of supported resource to filter query results.

Except for volume backups, compartments, and subnets, you can click the display name of the resource to view details. Query results are eventually consistent, but might not immediately include resources that you created very recently.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use the following operations to search for resources or find out what resources you can search for:

- SearchResources
- ListResourceTypes

Example: Finding Instance Resources With a Specific Defined Tag

This section describes how to use the API to query for a specific type of resource based on the resource's defined tags.

The following query will find instances with a defined tag within the namespace "rqs", where the tag's key is "costcenter" and the key's value is "1234".

```
query
 instance resources
 where
 (definedTags.namespace = 'rqs' && definedTags.key = 'costcenter' && definedTags.value = '1234')
```

When you use the SearchResources operation to issue the query, the request will look similar to the following. (This example purposefully omits the authorization header and other headers.)

```
POST /20180409/resources
Host: query.us-phoenix-1.oraclecloud.com
<authorization and other headers>
{
  "type": "Structured",
  "query": "query instance resources where (definedTags.namespace = 'rqs' && definedTags.key =
```

```
'costcenter' && definedTags.value = '1234')",
  "matchingContextType": "HIGHLIGHTS"
}
```

If your query produces results, the response will list the resources that match the resource type and tag that you specified. The response will look similar to the following:

```
{
  "items" : [ {
    "resourceType" : "Instance",
    "identifier" :
"ocid1.instance.oc1.phx.exampleawcbfhncvbh3siw2svbpgr3bopovy6hgnywfauxqo37ckdmr6hjya",
    "compartmentId" : "ocid1.tenancy.oc1..examplea46vssm7l5wsk5qa7cvbl63ctajep4bh6lv4vaifauxz6ec7jzg4q",
    "timeCreated" : "2018-10-31T22:48:47.855Z",
    "displayName" : "service-pkgs",
    "availabilityDomain" : "ABCd:PHX-AD-1",
    "lifecycleState" : "RUNNING",
    "freeformTags" : { },
    "definedTags" : {
      "rqs" : {
        "costcenter" : "1234"
      }
    },
    "searchContext" : null
  }, {
    "resourceType" : "Instance",
    "identifier" :
"ocid1.instance.oc1.phx.exampleanb3poce6z4omcvbzw66epp3pvbbww6hq7e2jfaux2lxvi3daxhra",
    "compartmentId" :
"ocid1.compartment.oc1..examplea43m3udlwrzwmbcvbk5hm3umk2khgfhjcgdttawjlfauxuqwsjiya",
    "timeCreated" : "2018-10-09T23:35:30.167Z",
    "displayName" : "prod-test",
    "availabilityDomain" : "ABCd:PHX-AD-2",
    "lifecycleState" : "RUNNING",
    "freeformTags" : { },
    "definedTags" : {
      "rqs" : {
        "costcenter" : "1234"
      }
    },
    "searchContext" : null
  }, {
    "resourceType" : "Instance",
```

```
    "identifier" :
"ocid1.instance.oc1.phx.examples7cz4z6b5hpdly2cvb56obhaiy4gvh2hdpz4akq4fauxpakvlqgya",
    "compartmentId" : "ocid1.tenancy.oc1..examplea46vssm7l5wsk5qa7cvbl63ctajep4bh6lv4fauxf4iz6ec7jzg4q",
    "timeCreated" : "2018-06-12T19:45:24.945Z",
    "displayName" : "BackupTest",
    "availabilityDomain" : "ABCd:PHX-AD-3",
    "lifecycleState" : "STOPPED",
    "freeformTags" : { },
    "definedTags" : {
      "rqs" : {
        "costcenter" : "1234"
      }
    },
    "searchContext" : null
  }, {
    "resourceType" : "Instance",
    "identifier" :
"ocid1.compartment.oc1..exampleaexfjsiad7gbi6r4hvmcvbk3a5hgkvutlswf54ulfauxks4p2jasq",
    "compartmentId" : "ocid1.tenancy.oc1..examplea46vssm7l5cvb5qa7gg5l63ctajep4bh6lv4fauxf4iz6ec7jzg4q",
    "timeCreated" : "2018-06-12T19:25:16.942Z",
    "displayName" : "personal_abc",
    "availabilityDomain" : "ABCd:PHX-AD-2",
    "lifecycleState" : "TERMINATED",
    "freeformTags" : { },
    "definedTags" : {
      "rqs" : {
        "costcenter" : "1234"
      }
    },
    "searchContext" : null
  }, {
    "resourceType" : "Instance",
    "identifier" :
"ocid1.compartment.oc1..examplealrskzczjqmrb3cvbj4yxdvqxahhffauxtu24tk5dhikoff4uliha",
    "compartmentId" : "ocid1.tenancy.oc1..examplea46vssm7l5wsk5qa7gg5l63cvbjep4bh6lv4fauxf4iz6ec7jzg4q",
    "timeCreated" : "2018-11-29T23:40:29.005Z",
    "displayName" : "test_unused",
    "availabilityDomain" : null,
    "lifecycleState" : "AVAILABLE",
    "freeformTags" : { },
    "definedTags" : {
      "rqs" : {
```

```
        "costcenter" : "1234"
      }
    },
    "searchContext" : null
  } ]
}
```

With these results, you can take additional action, if needed. For more information about a resource type, such as its attributes, see its reference page in the API Reference Guide. For the reference pages of resource types that have been indexed for Search, see Supported Resources.

Troubleshooting Search

This topic covers common issues related to Search and how you can address them:

- Query or Search Results are Not as Expected

Query or Search Results are Not as Expected

There are several reasons why you might not see results that you expect from a search or query.

Not all resource types have been indexed for Search. For a list of currently supported resource types, see Supported Resources.

You might not have the required permissions for the resource type that you want to view in search or query results. If there's no policy that grants you the permissions you need, then an administrator must create one for you or add you to a group that's already named in a policy. For more information, see Details for Search.

The query syntax you used might need adjustment. Verify that the conditions in your query language haven't restricted the results to a narrower set than you intended.

If you recently created a resource, it might not show up in search results immediately. Similarly, if you recently updated a resource, your changes might not immediately appear. At times, you might see a resource in a list view before you can see it in search results. The Search service is eventually consistent. Wait, and then try again.

CHAPTER 24 Security Guide and Announcements

This section of the Oracle Cloud Infrastructure documentation provides a guide to help you securely configure services and resources, and timely announcements relevant to emerging security issues.

- Oracle Cloud Infrastructure Security Guide
- Oracle Cloud Security Response to Intel L1TF Vulnerabilities
- Oracle Cloud Security Response to Intel Microarchitectural Data Sampling (MDS) Vulnerabilities

Oracle Cloud Infrastructure Security Guide

Oracle Cloud Infrastructure enables enterprises to maximize the number of mission-critical workloads that they can migrate to the cloud while continuing to maintain their desired security posture and reduce the overhead of building and operating data-center infrastructure. With Oracle Cloud Infrastructure, enterprise customers get unparalleled control of and transparency into their applications running in the cloud, including:

- Customer isolation that allows you to deploy your application and data assets in an environment that commits full isolation from other tenants and Oracle's staff, as well as between the same tenant's workloads.
- Always-on encryption that protects customer data at-rest and HTTPS-only public APIs.
- Easy-to-use security policy that allows you to constrain access to your services and segregate operational responsibilities to reduce risk associated with malicious and accidental user actions.
- Comprehensive log data that allows you to audit and monitor actions on your resources, helping you to meet your audit requirements while reducing security and operational risk.
- Identity federation that allows you to use your existing users and groups in the cloud.

- Support for bringing in third-party software solutions for protecting customer data and resources in the cloud.
- Fault-independent data centers that enable high availability scale-out architectures and are resilient against network attacks, ensuring constant uptime in the face of disaster and security attack.
- Rigorous internal processes and use of effective security controls in all phases of cloud service development and operation.
- Adherence to Oracle's strict security standards through third-party audits, certifications, and attestations. Oracle helps customers demonstrate compliance readiness to internal security and compliance teams, their customers, auditors, and regulators.

All of the Oracle Cloud Infrastructure security capabilities have been designed with one goal in mind: allowing you to run your mission-critical workloads in the cloud with complete control and confidence. Oracle continues to invest in the above areas and more to offer unmatched security and assurance to enterprise customers.

For an overview of Oracle Cloud Infrastructure's security, see Security Overview.

For service-specific best practices and policy examples, see Security Best Practices .

Security Overview

Oracle's mission is to build cloud infrastructure and platform services for your business to have effective and manageable security to run your mission-critical workloads and store your data with confidence.

Oracle Cloud Infrastructure's security approach is based on seven core pillars. Each pillar has multiple solutions designed to maximize the security and compliance of the platform.

CUSTOMER ISOLATION

Allow customers to deploy their application and data assets in an environment that commits full isolation from other tenants and Oracle's staff.

DATA ENCRYPTION

Protect customer data at-rest and in-transit in a way that allows customers to meet their security and compliance requirements for cryptographic algorithms and key management.

SECURITY CONTROLS

Offer customers effective and easy-to-use security management solutions that allow them to constrain access to their services and segregate operational responsibilities to reduce risk associated with malicious and accidental user actions.

VISIBILITY

Offer customers comprehensive log data and security analytics that they can use to audit and monitor actions on their resources, allowing them to meet their audit requirements and reduce security and operational risk.

SECURE HYBRID CLOUD

Enable customers to use their existing security assets, such as user accounts and policies, as well as third-party security solutions when accessing their cloud resources and securing their data and application assets in the cloud.

HIGH AVAILABILITY

Offer fault-independent data centers that enable high availability scale-out architectures and are resilient against network attacks, ensuring constant uptime in the face of disaster and security attack.

VERIFIABLY SECURE INFRASTRUCTURE

Follow rigorous processes and use effective security controls in all phases of cloud service development and operation. Demonstrate adherence to Oracle's strict security standards through third-party audits, certifications, and attestations. Help customers demonstrate compliance readiness to internal security and compliance teams, their customers, auditors, and regulators.

Also, Oracle employs some of the world's foremost security experts in information, database, application, infrastructure, and network security. By using Oracle Cloud Infrastructure, our customers directly benefit from Oracle's deep expertise and continuous investments in security.

Basic Security Considerations

The following principles are fundamental to using any application securely:

- Keep software up-to-date. This includes the latest product release and any patches that apply to it.
- Limit privileges as much as possible. Users should be given only the access necessary to perform their work. User privileges should be reviewed periodically to determine relevance to current work requirements.
- Monitor system activity. Establish who should access which system components, and how often, and monitor those components.
- Learn about and use the Oracle Cloud Infrastructure security features. For more information, see Security Services and Features.
- Use secure best practices. For more information, see Security Best Practices .
- Keep up-to-date on security information. Oracle regularly issues security-related patch updates and security alerts. Install all security patches as soon as possible. See the Critical Patch Updates and Security Alerts website.

Understanding the Oracle Cloud Infrastructure Environment

When planning your Oracle Cloud Infrastructure deployment, consider the following:

WHICH RESOURCES MUST BE PROTECTED?

- Protect customer data, such as credit card numbers.
- Protect internal data, such as proprietary source code.
- Protect system components from being disabled by external attacks or intentional system overloads.

WHO ARE YOU PROTECTING DATA FROM?

For example, you must protect your subscribers' data from other subscribers, but someone in your organization needs to access that data to manage it. Analyze your workflows to determine who needs access to the data. Consider carefully how much access to give a

system administrator; it is possible that a system administrator can manage your system components without needing to access the system data.

WHAT WILL HAPPEN IF PROTECTIONS ON A STRATEGIC RESOURCE FAIL?

Sometimes, a fault in your security scheme is nothing more than an inconvenience. In other cases, a fault might cause great damage to you or your customers. Understanding the security ramifications of each resource will help you protect it properly.

Shared Security Model

Oracle Cloud Infrastructure offers best-in-class security technology and operational processes to secure its enterprise cloud services. However, for you to securely run your workloads in Oracle Cloud Infrastructure, you must be aware of your security and compliance responsibilities. By design, Oracle provides security of cloud infrastructure and operations (cloud operator access controls, infrastructure security patching, and so on), and you are responsible for securely configuring your cloud resources. Security in the cloud is a shared responsibility between you and Oracle.

In a shared, multi-tenant compute environment, Oracle is responsible for the security of the underlying cloud infrastructure (such as data-center facilities, and hardware and software systems) and you are responsible for securing your workloads and configuring your services (such as compute, network, storage, and database) securely.

In a fully isolated, single-tenant, bare metal server with no Oracle software on it, your responsibility increases as you bring the entire software stack (operating systems and above) on which you deploy your applications. In this environment, you are responsible for securing your workloads, and configuring your services (compute, network, storage, database) securely, and ensuring that the software components that you run on the bare metal servers are configured, deployed, and managed securely.

More specifically, your and Oracle's responsibilities can be divided into the following areas:

- **Identity and Access Management (IAM):** As with all Oracle cloud services, you should protect your cloud access credentials and set up individual user accounts. You are responsible for managing and reviewing access for your own employee accounts

and for all activities that occur under your tenancy. Oracle is responsible for providing effective IAM services such as identity management, authentication, authorization, and auditing.

- **Workload Security:** You are responsible for protecting and securing the operating system and application layers of your compute instances from attacks and compromises. This protection includes patching applications and operating systems, operating system configuration, and protection against malware and network attacks. Oracle is responsible for providing secure images that are hardened and have the latest patches. Also, Oracle makes it simple for you to bring the same third-party security solutions that you use today.
- **Data Classification and Compliance:** You are responsible for correctly classifying and labeling your data and meeting any compliance obligations. Also, you are responsible for auditing your solutions to ensure that they meet your compliance obligations.
- **Host Infrastructure Security:** You are responsible for securely configuring and managing your compute (virtual hosts, containers), storage (object, local storage, block volumes), and platform (database configuration) services. Oracle has a shared responsibility with you to ensure that the service is optimally configured and secured. This responsibility includes hypervisor security and the configuration of the permissions and network access controls required to ensure that hosts can communicate correctly and that devices are able to attach or mount the correct storage devices.
- **Network Security:** You are responsible for securely configuring network elements such as virtual networking, load balancing, DNS, and gateways. Oracle is responsible for providing a secure network infrastructure.
- **Client and Endpoint Protection:** Your enterprise uses various hardware and software systems, such as mobile devices and browsers, to access your cloud resources. You are responsible for securing all clients and endpoints that you allow to access Oracle Cloud Infrastructure services.
- **Physical Security:** Oracle is responsible for protecting the global infrastructure that runs all of the services offered in Oracle Cloud Infrastructure. This infrastructure

consists of the hardware, software, networking, and facilities that run Oracle Cloud Infrastructure services.

For information about using security credentials to access Oracle Cloud Infrastructure, see Security Credentials.

Infrastructure Security

Our security model is built around people, process, tooling, and a common security "platform" of methodologies and approaches from which we build our products. We apply this model to our core security components of Security Culture, Security Design and Controls, Secure Software Development, Personnel Security, Physical Security, and Security Operations that we use to protect and secure our customers and business.

Security Culture

We believe that a dynamic security-first culture is vital to building a successful security-minded organization. We have cultivated a holistic approach to security culture in which all our team members internalize the role that security plays in our business and are actively engaged in managing and improving our products' security posture. We have also implemented mechanisms that assist us in creating and maintaining a security-aware culture.

- **Security-minded leadership:** Our senior leadership is actively involved in our security planning, monitoring and management. We define and measure ourselves against security metrics and include security as a component of our team evaluation processes.
- **Embedded expertise:** To help with driving security practices within our team, we have an embedded security-engineering model with security team members sitting and working with our product development teams. This approach enables our security organization to build deep understanding of the product-development processes and system architectures. We are also able to better assist teams in solving security challenges in real time and drive security initiatives more effectively.
- **Common security standards:** We actively work to integrate security into our products and operations. One way we have done this is to establish a security standards baseline. Our objective in creating this baseline is to provide a single security point of

reference for business that establishes clear and actionable guidelines. The security baseline is updated frequently to incorporate learned lessons and reflect emerging business factors. We have also created a series of support materials to assist our teams in implementing security controls including reference architectures, implementation guides, and access to security experts.

- **Values of openness, constructive debate, and encouraged escalation:** Security issues can be addressed only when the people who can fix them are aware of them. We believe that openness and transparency, constructive debate, and encouraged escalation make us stronger. We encourage escalation, and we work to create an environment where raising issues early and often is rewarded.
- **Security training awareness:** We maintain robust security and awareness training programs that raise awareness and reinforce our security culture. We require in-depth security training sessions for all new employees as well as annual refresher trainings, and we provide security training that is tailored to our employees' specific job roles. All our software developers undergo a secure development training that establishes baseline security requirements for product development and provides best practices. We also work to provide engaging and innovative forms of security awareness training such as guest speakers and interactive forums (and we're not above providing food, drinks, or swag to drive attendance).

Security Designs and Controls

Security is integrated into our products and operations through our Oracle Cloud Infrastructure Methodology. This centralized methodology defines our approach for the core security areas that form the security foundation from which we build our products. This approach lends itself to agility and helps us apply best practices and lessons learned from one product across the business, thus raising the security of all our products.

- **User authentication and access control:** Least-privilege access is used to grant access to production systems, and the approved lists of service team members are periodically reviewed to revoke access when there is no justifiable need. Access to production environments requires multi-factor authentication (MFA). The MFA tokens are granted by the security team, and tokens of inactive members are disabled. All access to production systems is logged, and the logs are stored for security analysis.

- **Change management:**Oracle Cloud Infrastructure follows a defined and rigorous change management and deployment process that uses purpose-built proprietary testing and deployment tools. All changes deployed into our production environment follow a testing and approval process prior to release. This process is designed to ensure that changes operate as intended, and can otherwise be rolled back to a previous known good state to recover gracefully from unforeseen bugs or operational issues. We also track the integrity of critical system configurations to ensure that they align with expected state.
- **Vulnerability management:** We use both internal penetration testing teams and external industry experts to help us identify potential vulnerabilities in our products. These exercises help us improve the security of our products, and we work to incorporate the lessons that we learn into our future development work. Oracle Cloud Infrastructure hosts undergo periodic vulnerability scanning using industry-standard scanners. Scan results are triaged to validate applicability of findings to the Oracle Cloud Infrastructure environment, and that applicable findings are patched by our product teams.
- **Incident response:** We have developed strong processes and mechanisms to enable us to respond to and address incidents as they arise. We maintain 24/7 incident response teams ready to detect and respond to events. Our critical staff members carry paging devices that enable us to call on the expertise needed to bring issues to resolution. We have also built a process to help us learn from our incidents. We perform root cause analysis through our Corrective Action/Preventative Action (CAPA) process. CAPAs are intended to discover process gaps and changes that should be made by the business after an incident. CAPAs act as a common language that we can use to reflect on an issue and capture concrete steps to improve future operational readiness. CAPAs capture the root cause of an issue, what is required to contain or fix the issue, and what steps we must take to ensure that the issue does not recur. Our leadership team reviews all CAPAs, looks for cross-organizational applications for learned lessons, and ensures that actions are implemented in a timely manner.
- **Security logging and monitoring:** We have created automated mechanisms to log various security-relevant events (for example, API calls and network events) in the

infrastructure, and monitor the logs for anomalous behavior. Alerts generated by monitoring mechanisms are tracked and triaged by the security team.

- Network security: By default, customer communications with Oracle Cloud Infrastructure services are done using the latest TLS ciphers and configuration to secure customer data in transit, and hinder any man-in-the-middle attacks. As a further defense in depth, customer commands to the services are digitally signed using public keys, to prevent any tampering. The services also deploy proven, industry-leading tools and mechanisms to mitigate distributed denial of service (DDoS) attacks and maintain high availability.
- **Control plane security:**Oracle Cloud Infrastructure back-end (control plane) hosts are securely isolated from customer instances by using network ACLs. Provisioning and management of customer instances are done by software agents that must interact with the backend hosts. Only authenticated and authorized software agents can successfully interact with Oracle Cloud Infrastructure back-end hosts. For back-end hosts, pre-production environments (for example, dev, test, and integ) are separated from production environments so that any development and test activities do not have any impact on production systems.
- **Server security and media management:** Oracle has a long history of enterprise-class secure hardware development. Our Hardware Security team is responsible for designing and testing the security of the hardware used to deliver Oracle Cloud Infrastructure services. This team works with our supply chain and tests hardware components to validate them against rigorous Oracle Cloud Infrastructure hardware security standards. This team also works closely with our product development functions to ensure that hardware can be returned to a pristine, safe state after being released by customers.
- **Secure host wipe and media destruction:**Oracle Cloud Infrastructure instances are securely wiped after hardware is released by customers. This secure wipe restores hardware to a pristine state. We have re-engineered the platform with proprietary hardware components that allow us to wipe and reinitialize the hardware in a secure manner. When the underlying hardware has reached end-of-life, it is securely destroyed. Before leaving our data centers, drives are rendered unusable by using industry-leading media destruction devices.

Secure Software Development

Secure product development requires consistently applied methodologies that conform to clear security objectives and principles. We build security practices into every element of our product development life cycle. Oracle employs formal secure product development standards that are a roadmap and guide for developers. These standards discuss general security knowledge areas such as design principles and common vulnerabilities, and provide specific guidance on topics such as data validation, data privacy, and user management.

Oracle secure product development standards have evolved and expanded over time to address the common issues affecting code, new threats as they are discovered, and new use cases by Oracle customers. The standards incorporate insights and learned lessons; they do not live in a vacuum, nor are they an "after the fact" addendum to software development. They are integral to language-specific standards such as C/C++, Java, PL/SQL, and others, and are a cornerstone to Oracle's secure development programs and processes.

Security assurance analysis and testing verify security qualities of Oracle products against various types of attacks. There are two broad categories of tests employed for testing Oracle products: static and dynamic analysis. These tests fit differently in the product development lifecycle and tend to find different categories of issues, so they are used together by Oracle product teams.

Personnel Security

Our people make our business. We strive to hire the best, and we invest in and continue to develop our employees. We value training, and we require not only baseline security training for all our employees but also specialized training to keep our teams abreast of the latest security technologies, exploits, and methodologies. In addition to standard annual corporate training programs that cover our information security and privacy programs (among many others), we engage with a broad spectrum of industry groups and send our employees to specialist conferences to collaborate with other industry experts on emerging challenges. The objectives of our security training programs are to help our employees better protect our customers and products, to enable employees to grow in their knowledge areas around security, and to further our mission to attract and retain the best talent.

We work to recruit the best talent for our team as we grow, and we hire people with strong ethics and good judgment. All our employees undergo pre-employment screening as

permitted by law, including criminal background checks and prior-employment validation. We also maintain performance evaluation processes to recognize good performance and help our teams and employees identify opportunities for growth. We maintain both team and employee evaluation processes, and we use security as a component of our team evaluation processes. This approach provides our teams and leadership visibility into how our teams are performing against our security standards and enables us to identify best practices and improvement areas for critical security processes.

PHYSICAL SECURITY

Oracle Cloud Infrastructure data centers are designed for security and availability of customer data. This approach begins with our site selection process. Candidate build sites and provider locations undergo an extensive risk evaluation process that considers environmental threats, power availability and stability, vendor reputation and history, neighboring facility functions (for example, high-risk manufacturing or high-threat targets), and geopolitical considerations, among other criteria.

Oracle Cloud Infrastructure data centers align with Uptime Institute and Telecommunications Industry Association (TIA) ANSI/TIA-942-A Tier 3 or Tier 4 standards and follow a N2 redundancy methodology for critical equipment operation. Data centers housing Oracle Cloud Infrastructure services use redundant power sources and maintain generator backups in case of widespread electrical outage. Server rooms are closely monitored for air temperature and humidity, and fire suppression systems are in place. Data center staff are trained in incident response and escalation procedures to address security or availability events that may arise.

We take a layered approach to physical security that starts with the site build. Oracle Cloud Infrastructure data center facilities are durably built with steel, concrete, or comparable materials and are designed to withstand impact from a light vehicle strike. Our sites are staffed with security guards who are ready to respond to incidents 24 hours a day, 7 days a week, 365 days a year. The exterior of the sites is secured with perimeter barriers and vehicle checks are actively monitored by a guard force and cameras that cover the building perimeter.

All persons entering our data centers must first go through a layer of security at the site entrances, which are staffed with security guards. Persons without site-specific security badges entering the site must present government-issued identification and have an approved

access request granting them access to the data center building. All employees and visitors must wear visible, official identification badges at all times. There are additional security layers between the entrance and server rooms that vary depending on the site build and risk profile. Data center server rooms are built with additional security layers including cameras that cover server rooms, two-factor access control, and intrusion-detection mechanisms. Physical barriers are in place to create isolated security zones around server and networking racks that span from the floor (including below the raised floor where applicable) to the ceiling (including above ceiling tiles where applicable).

Access to Oracle Cloud Infrastructure data centers is carefully controlled and follows a least-privilege access approach. All access to server rooms must be approved by authorized personnel and is granted only for the necessary period. Access usage is audited, and access provisioned within the system is periodically reviewed by data-center leadership. Server rooms are isolated into secure zones that are managed on a zone-by-zone basis, and access is provisioned only for those zones required by personnel.

Security Operations

The Oracle Cloud Infrastructure Security Operations team is responsible for monitoring and securing the unique Oracle Cloud Infrastructure hosting and virtual networking technologies. The team works and trains directly with the Oracle engineers who develop these technologies to leverage the unique security and introspection capabilities they provide.

We monitor emerging internet security threats daily and implement appropriate response and defense plans to address risks to the business. When we determine that urgent changes are recommended that are within the scope of the customers' responsibilities, we issue security alert bulletins to those customers to ensure their protection.

In the case of a detected or reported security issue that affects Oracle Cloud Infrastructure servers or networks, Security Operations staff is available 24/7 to respond, escalate, or take required corrective action. When necessary, we will escalate and coordinate with external parties (including network and hosting service providers, hardware vendors, or law enforcement) to protect Oracle Cloud Infrastructure, our customers, and our network's security and reputation.

All actions performed in response to a security issue by the Security Operations team are done according to our documented process, and are logged in accordance with compliance

requirements. Care is always taken to protect the goals of service and data integrity, privacy, and business continuity.

Customer Data Protection

Data Rights and Ownership

Oracle Cloud Infrastructure customers retain all ownership and intellectual property rights in and to their content. Customer data protection is critically important, and we strive to be transparent with our data protection processes as well as law enforcement requests that we might receive.

Data Privacy

Oracle complies with the EU-U.S. Privacy Shield Framework as set forth by the U.S. Department of Commerce regarding the collection, use, and retention of personal information transferred from the European Union to the United States. Oracle is also responsible for ensuring that third parties who act as an agent on our behalf do the same.

Oracle has certified to the Department of Commerce that it adheres to the Privacy Shield Principles. If there is any conflict between the terms in our privacy policy and the Privacy Shield Principles, the Privacy Shield Principles shall govern. To learn more about the Privacy Shield program, and to view our certification, visit https://www.privacyshield.gov/list.

For personal information received or transferred pursuant to the Privacy Shield Framework, Oracle is subject to the regulatory enforcement powers of the U.S. Federal Trade Commission.

Oracle continues to adhere to the underlying European privacy principles of the U.S.-Swiss Safe Harbor for the processing of Personal Information received from Switzerland. To learn more about the Safe Harbor program, and to view our certification, visit https://safeharbor.export.gov/swisslist.aspx.

Law Enforcement Requests

Except as otherwise required by law, Oracle will promptly notify customers of any subpoena, judicial, administrative or arbitral order of an executive or administrative agency or other governmental authority that it receives and which relates to the personal data Oracle is

processing on the customer's behalf. Upon customer request, Oracle will provide customers with reasonable information in its possession relevant to the law enforcement request and any assistance reasonably required for them to respond to the request in a timely manner.

COMPLIANCE

Oracle Cloud Infrastructure is built for enterprises. We operate under practices aligned with the ISO/IEC 27002 Code of Practice for information security controls, from which we have identified a comprehensive set of security controls that apply to our business. Oracle Cloud Infrastructure is still a new product line, and we must operate for a period of time in order for these security controls and our operations to undergo external audit. As an enterprise cloud, we plan to pursue a broad suite of industry and government certifications, audits, and regulatory programs.

Security Services and Features

A key objective of Oracle Cloud Infrastructure is to allow you to create a logical extension of your on-premises infrastructure and data centers in Oracle Cloud Infrastructure. You can gain the benefits of a modern public cloud without having to compromise or reinvent your existing security posture. This idea was central to the design of all our infrastructure and services.

Regions and Availability Domains

To provide data availability and durability, Oracle Cloud Infrastructure enables you to select from infrastructure with distinct geographic and threat profiles. A region is the top-level component of the infrastructure. Each region is a separate geographic area with multiple, fault-isolated locations called availability domains. An availability domain is the subcomponent of a region and is independent and highly reliable. Each availability domain is built with fully independent infrastructure: buildings, power generators, cooling equipment, and network connectivity. With physical separation comes protection against natural and other disasters. Availability domains within the same region are connected by a secure, high-speed, low-latency network, which allows customers to build and run highly reliable applications and workloads with minimum impact to application latency and performance. All links between availability domains are encrypted. Each region has one or more availability domains, each allowing customers to deploy highly available applications.

Identity and Access Management (IAM) Service

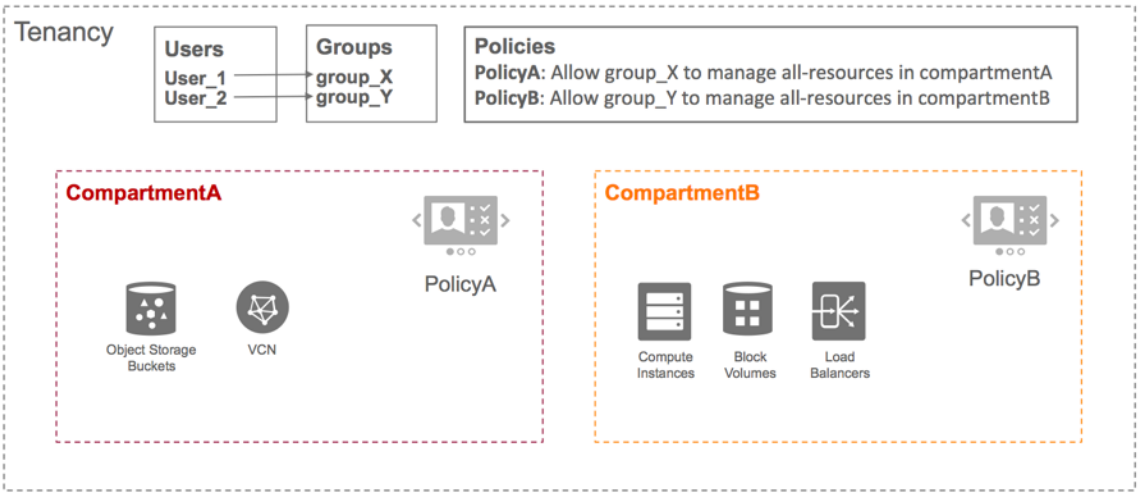
The Oracle Cloud Infrastructure Identity and Access Management (IAM) service is built to meet the requirements of enterprises, and it provides authentication and authorization for all their Oracle Cloud Infrastructure resources and services. An enterprise can use a single tenancy shared by various business units, teams, and individuals while maintaining security, isolation, and governance.

When a customer joins Oracle Cloud Infrastructure, a tenancy is created. A tenancy is a virtual construct that contains all of the Oracle Cloud Infrastructure resources that belong to the customer. The administrator of the tenancy can create users and groups and assign them least-privileged access to resources that are partitioned into compartments. A compartment is a group of resources that can be managed as a single logical unit, providing a streamlined way to manage large infrastructure. For example, a customer can create a compartment (`HR-Compartment`) to host a specific set of cloud network, compute instances, and storage volumes necessary to host its HR applications. Compartments are a fundamental component of Oracle Cloud Infrastructure for organizing and isolating cloud resources. Customers use them to clearly separate resources for the purposes of isolation (separating the resources for one project or business unit from another). A common approach is to create a compartment for each major part of an organization. Unlike most Oracle Cloud Infrastructure services that are regionally scoped, the IAM service resources are global. Customers can have a single tenancy across multiple regions.

The following are key IAM primitives:

- **Resource:** A cloud object that a company's employees create and use when interacting with Oracle Cloud Infrastructure services, for example, compute instances, block storage volumes, virtual cloud networks (VCNs), subnets, and route tables.
- **Policy:** A set of authorization rules that define access to resources within a tenancy.
- **Compartment:** A heterogeneous collection of resources for the purposes of security isolation and access control.
- **Tenancy:** The root compartment that contains all of an organization's resources. Within a tenancy, administrators can create one or more compartments, create more users and groups, and assign policies that grant groups the ability to use resources within a compartment.

- **User:** A human being or system that needs access to manage their resources. Users must be added to groups in order to access resources. Users have one or more credentials that must be used to authenticate to Oracle Cloud Infrastructure services. Federated users are also supported.
- **Group:** A collection of users who share a similar set of access privileges. Administrators can grant access policies that authorize a group to consume or manage resources within a tenancy. All users in a group inherit the same set of privileges.
- **Identity Provider:** A trusted relationship with a federated identity provider. Federated users who attempt to authenticate to the Oracle Cloud Infrastructure console are redirected to the configured identity provider. After successfully authenticating, federated users can manage Oracle Cloud Infrastructure resources in the console just like a native IAM user. Currently, Oracle Cloud Infrastructure supports the Oracle Identity Cloud Service and Microsoft Active Directory Federation Service (ADFS) as identity providers. Federated groups are mapped to native IAM groups to define the policies apply to a federated user.



All customer calls to access Oracle Cloud Infrastructure resources are first authenticated by the IAM service (or federated provider) and then authorized based on IAM policies. A customer can create a policy that gives a set of users permission to access the infrastructure resources (network, compute, storage, and so on) within a compartment in the tenancy.

These policies are flexible and are written in a human-readable form that is easy to understand and audit. A policy contains one or more policy statements that follow this easy-to-understand syntax:

```
Allow group <group_name> to <verb> <resource-type> in compartment <compartment_name>
```

A verb defines the type of access covered. Oracle defines the following verbs that you can use in your policy statements:

- **inspect:** Provides the ability to list resources, without access to any confidential information or user-specified metadata that might be part of that resource.
- **read:** Includes inspect plus the ability to get user-specified metadata and the actual resource itself.
- **use:** Includes read plus the ability to work with existing resources (the actions vary by resource type). Includes the ability to update the resource, except for resource types where the update operation has the same effective impact as the create operation (for example, `UpdatePolicy` and `UpdateSecurityList`). In such cases, the update ability is available only with the manage verb. In general, this verb does not include the ability to create or delete that type of resource.
- **manage:** Includes all permissions for the resource.

The following example policy enables the `GroupAdmins` group to create, update, or delete any groups:

```
Allow group GroupAdmins to manage groups in tenancy
```

Each user has one or more of the following credentials to authenticate themselves to Oracle Cloud Infrastructure. Users can generate and rotate their own credentials. In addition, a tenancy security administrator can reset credentials for any user within their tenancy.

- **Console password:** Used to authenticate a user to the Oracle Cloud Infrastructure Console.
- **API key:** All API calls are signed using a user-specific 2048-bit RSA private key. The user creates a public key pair, and uploads the public key in the Console.
- **Auth token:** Auth tokens are Oracle-generated token strings that you can use to authenticate with third-party APIs that do no support Oracle Cloud Infrastructure's

signature-based authentication. For example, use an auth token to authenticate with a Swift client. To ensure sufficient complexity, the token is created by the IAM service and cannot be provided by a customer.

- **Customer secret key:** Used by Amazon S3 clients to access the Object Storage service's S3-compatible API. To ensure sufficient complexity, the password is created by the IAM service and cannot be provided by a customer.

Audit Service

The Oracle Cloud Infrastructure Audit service records all API calls to resources in a customer's tenancy as well as login activity from the graphical management Console. Using the Audit service, customers can achieve their own security and compliance goals by monitoring all user activity within their tenancy. Because all Console, SDK, and command line (CLI) calls go through our APIs, all activity from those sources is included. Audit records are available through an authenticated, filterable query API or can be retrieved as batched files from Oracle Cloud Infrastructure Object Storage. Audit log contents include what activity occurred, the user that initiated it, the date and time of the request, as well as source IP, user agent, and HTTP headers of the request.

Compute Service

Compute is a core component of Oracle Cloud Infrastructure and provides on-demand and elastic compute capabilities with enterprise-grade security and performance. Customers can provision thousands of compute instances and scale them up or down through an easy-to-use web-based management console. Programmatic support to do the same is available through feature-rich SDKs and command-line interfaces (CLIs). All compute instances are hosted in Oracle enterprise-grade data centers.

Compute instances are based on high-performance server hardware that uses latest-generation, multi-core server CPUs, large amounts of memory, and high-throughput NVMe local storage. Oracle Cloud Infrastructure provides bare metal (BM) and virtual machine (VM) instances. Customers can choose instances that fit their performance, cost, and software flexibility requirements.

- **Bare metal instances:** In bare metal instances, physical servers are dedicated to a single customer who has complete control over the server. There is no Oracle-managed hypervisor and Oracle personnel have no access to memory or local (NVMe) storage while the instance is running. All network virtualization is performed off-box and only the Oracle Integrated Lights Out Manager (ILOM) is accessible to the infrastructure (required in order to remotely reboot or reprovision instances). These bare metal instances offer consistent high performance and are immune to any noisy-neighbor issues. Customers have OS-level administrative privileges to the bare metal instance. After a customer terminates their bare metal instance, the server undergoes an automated disk and firmware-level wipe process to ensure isolation between customers.
- **Virtual machine (VM) instances:** Customers with flexibility requirements or those who don't need a dedicated bare metal instance can opt for VMs. Multi-tenant customer VMs in Oracle Cloud Infrastructure are managed by a security hardened hypervisor that provides strong isolation between customers.

Oracle Cloud Infrastructure instances use key-based SSH by default. Customers provide the SSH public keys to Oracle Cloud Infrastructure and securely use the SSH private keys for accessing the instances. Oracle recommends using key-based SSH to access Oracle Cloud Infrastructure instances. Password-based SSH could be susceptible to brute-forcing attacks, and are not recommended.

Oracle Linux images hardened with the latest security updates are available for you to run on Oracle Cloud Infrastructure instances. Oracle Linux images run the Unbreakable Enterprise Kernel (UEK) and support advanced security features such as Ksplice to apply security patches without booting, which allows enterprises to live-update their instances without any disruption. In addition to Oracle Linux, Oracle Cloud Infrastructure makes a growing list of other OS images available, including CentOS, Ubuntu, and Windows Server. You can also bring your own custom images. All Oracle-provided images come with secure defaults including OS-level firewalls turned on by default.

Networking Service

High-throughput and reliable networking is fundamental to public-cloud infrastructure that delivers compute and storage services at scale. As a result, we invested significant innovation

in Oracle Cloud Infrastructure networking to support requirements of enterprise customers and their workloads. Oracle Cloud Infrastructure regions have been built with a state-of-the-art, non-blocking Clos network that is not over-subscribed and provides customers with a predictable, high-bandwidth, low latency network. The data centers in a region are networked to be highly available and have low-latency connectivity between them.

The Oracle Cloud Infrastructure Networking service offers a customizable private network (a VCN, or virtual cloud network) to customers, which enforces logical isolation of customer Oracle Cloud Infrastructure resources. As with their on-premises network in their data centers, customers can set up a VCN with hosts with private IP addresses, subnets, route tables and gateways using VCN. The VCN can be configured for internet connectivity, or connected to the customer's private data center through an IPSec VPN gateway or FastConnect. FastConnect offers a private connection between an existing network's edge router and Dynamic Routing Gateways (DRG). Traffic does not traverse the internet.

The Networking service also supports bi-directional, stateful and stateless firewalls that allow customers to initialize network security access controls. Firewalls and ACLs specified for a customer VCN are propagated throughout the network topology and control plane, ensuring a multi-tiered and defense-in-depth implementation. Each tenant (customer) can create multiple VCNs to implement logical grouping of their resources.

The following are key networking concepts associated with a VCN:

- **Subnets:** The primary subdivision of a VCN. Subnets have historically been specific to an availability domain, but can now be regional (covering all availability domains in the region). Subnets can be marked as private upon creation, which prevents instances launched in that subnet from having public IP addresses.
- **Internet gateway:** A virtual router that provides public internet connectivity from a VCN. By default, a newly created VCN has no internet connectivity.
- **Dynamic routing gateway:** A virtual router that provides a path for private traffic between a VCN and a data center’s network. It is used with an IPSec VPN or Oracle Cloud Infrastructure FastConnect connection to establish private connectivity between a VCN and an on-premises or other cloud network.

- **NAT gateway:** A virtual router that gives cloud resources without public IP addresses access to the internet without exposing those resources to incoming internet connections.
- **Service gateway:** A virtual router that gives cloud resources private access to Oracle services such as Object Storage without using an internet gateway or NAT gateway.
- **Routing tables:** Virtual routing tables that give the subnets access to the VCN's gateways (Internet Gateway and Dynamic Routing Gateway). Routes can also use private IPs as a target to implement network functionality such as NAT, firewalls, IDS, and so on.
- **Primary VNICs:** Subnets contain virtual network interface cards (VNICs) that attach to instances. The VNIC determines how the instance connects with endpoints inside and outside the VCN. Each instance has a primary VNIC that is created during instance launch and cannot be removed. During instance launch, the Networking service also assigns a public IP address. Customers can override that behavior during instance launch and request to have no public IP address assigned.
- **Secondary VNICs:** VNICs with public and private IP addresses that can be attached to an instance. In a bring-your-own-hypervisor (BYOH) scenario, where customers can run their hypervisor on a BM instance, a secondary VNIC can be assigned to a VM, to allow VCN networking for the VM. This is very useful for running virtual security appliances in a VCN.
- **IPSec VPN connection:** A secure VPN connection between a VCN and a data center.
- **Security lists:** Virtual firewall rules that define allowed ingress and egress to an instance at the packet level. Individual rules can be defined to be stateful or stateless.

Virtual firewalls are implemented by using VCN security lists. Customers can specify a set of firewall rules and associate them with one or more subnets. Associating a security list with a subnet applies those firewall rules to all instances running inside the subnet, at the packet level. There are two types of firewall rules:

- **Ingress rules:** Ingress rules specify the source (IP CIDR and port range), destination port range, and protocol to match on, and are applied to ingress network connections.

- **Egress rules:** Egress rules specify the destination (IP CIDR and port range), source port range, and protocol to match on, and are applied to egress network connections.

Every VCN has a default security list customers may optionally use that allows only SSH and certain types of important ICMP ingress traffic, and all egress traffic. Customers can associate multiple security lists with a subnet. The subnet uses the default security list if the customer doesn't specify another list for the subnet to use.

For further information about security in the Networking service, see:

- Ways to Secure Your Network
- Access Control
- Security Lists

Storage Services

Oracle Cloud Infrastructure offers multiple storage solutions to meet the performance and durability requirements of customers:

- Local Storage: NVMe-backed storage on compute instances, offering extremely high IOPS.
- Block Volumes: Network-attached storage volumes, attachable to compute instances.
- Object Storage: Regional service for storing large amounts of data as objects, providing strong consistency and durability.

The Oracle Cloud Infrastructure Block Volumes service provides persistent storage that can be attached to compute instances using the iSCSI protocol. The volumes are stored in high-performance network storage and support automated backup and snapshot capabilities. Volumes and their backups are accessible only from within a customer's VCN and are encrypted at rest using unique keys. For additional security, iSCSI CHAP authentication can be required on a per-volume basis.

The Oracle Cloud Infrastructure Object Storage service provides highly scalable, strongly consistent, and durable storage for objects. API calls over HTTPS provide high-throughput access to data. All objects are encrypted at rest using unique keys. Objects are organized by

bucket, and, by default, access to buckets and objects within them requires authentication. Users can use IAM security policies to grant users and groups access privileges to buckets.

To allow bucket access by users who do not have IAM credentials, the bucket owner (or a user with necessary privileges) can create pre-authenticated requests that allow authorized actions on buckets or objects for a specified duration. Alternately, buckets can be made public, which allows unauthenticated and anonymous access. Given the security risk of inadvertent information disclosure, Oracle highly recommends carefully considering the business case for making buckets public. Object Storage enables you to verify that an object was not unintentionally corrupted by allowing an MD5 hash to be sent with the object (or with each part, in the case of multipart uploads) and returned upon successful upload. This hash can be used to validate the integrity of the object.

In addition to its native API, the Object Storage service supports Amazon S3 compatible APIs. Using the Amazon S3 Compatibility API, customers can continue to use the existing Amazon S3 tools (for example, SDK clients), and partners can modify their applications to work with Object Storage, with minimal changes to their applications. Their native API can co-exist with the Amazon S3 Compatibility API, which supports CRUD operations. Before customers can use the Amazon S3 Compatibility API, they must create an S3 Compatibility API key. After they've generated the necessary key, they can use the Amazon S3 Compatibility API to access Object Storage in Oracle Cloud Infrastructure.

Database Service

Oracle Cloud Infrastructure makes it easy to run, scale, and secure your Oracle databases (DBs) in the cloud. The Oracle Cloud Infrastructure Database service offers three types of DB systems:

- Bare metal: Comprising 1-node DB and 2-node Real Application Cluster (RAC) systems, providing exceptional performance at cost-effective pricing.
- Exadata: Proven industry-leading Exadata DB systems in quarter, half, and full rack configurations.
- Virtual machine: Allows customers to create full-featured Oracle databases on VM shapes with various cores.

DB systems are accessible only from a customer's VCN, and customers can configure VCN security lists to control network access to their databases. The Database service is integrated with Oracle Cloud Infrastructure IAM for controlling which users can launch and manage DB systems. By default, data is encrypted at rest using Oracle TDE with master keys stored in an Oracle Wallet on each DB system. RMAN backups of DB systems are encrypted and stored in customer-owned buckets in the Object Storage service. Customers need to create a bucket for DB backups and configure the Oracle Database Cloud Backup module with an auth token (to use with the Swift API) and IAM permissions to access the bucket. Alternately, DB backups can be made to local NVMe storage on the DB system.

Each user automatically has the ability to create, update, and delete their own auth tokens in the Console or the API. An administrator does not need to create a policy to give a user those abilities. Administrators (or anyone with permission to the tenancy) also have the ability to manage auth tokens for other users. Any user of a Swift client that integrates with Object Storage needs permission to work with the service.

Load Balancing Service

Oracle Cloud Infrastructure Load Balancing provides automated traffic distribution to compute instances in a customer's VCN. Load balancers (LBs) can be created as public (accepting traffic from the internet and directing it to private instances) or private (directing traffic between private instances). LBs can be configured for SSL termination using customer-provided certificates; end-to-end SSL, whereby the LB terminates the SSL connection and creates a new SSL connection to the backend; or SSL tunneling, in which the SSL connection is passed through to the backend (TCP load balances only). The Load Balancing service supports TLS 1.2 by default, and prioritizes the following forward-secrecy ciphers in the TLS cipher-suite:

- ECDHE-RSA-AES256-GCM-SHA384
- ECDHE-RSA-AES256-SHA384
- ECDHE-RSA-AES128-GCM-SHA256
- ECDHE-RSA-AES128-SHA256
- DHE-RSA-AES256-GCM-SHA384
- DHE-RSA-AES256-SHA256

- DHE-RSA-AES128-GCM-SHA256
- DHE-RSA-AES128-SHA256

Managed Domain Name Servers (DNS) Service

The Oracle Cloud Infrastructure DNS service provides dynamic, static, and recursive DNS solutions for enterprise customers. The service connects visitors to customer websites and applications with fast and secure services. The DNS service operates on a global anycast network with 18 points of presence (PoPs) on five continents and offers fully redundant DNS constellations and multiple Tier 1 transit providers per PoP. The solution provides a DNS-based Distributed Denial of Services (DDoS) protection and in-house security expertise that leverages a vast sensor network that collects and analyzes over 240 billion data points per day. The DNS service also fully supports the secondary DNS features to complement the customer's existing DNS service, providing resiliency at the DNS layer.

Oracle Cloud Security Testing Policy

This section describes the Oracle Cloud Security Testing policy and how you can submit a request to schedule the tests of your Oracle Cloud services. The Oracle Cloud Security Testing Policy describes when and how you may conduct certain types of security testing of Oracle Cloud Services, including vulnerability and penetration tests, as well as tests involving data scraping tools. Notwithstanding anything to the contrary, any such testing of Oracle Cloud Services may be conducted only by customers who have an Oracle Account with the necessary privileges to file service maintenance requests, and who are signed-in to the environment that will be the subject of such testing.

Penetration and Vulnerability Testing

Oracle regularly performs penetration and vulnerability testing and security assessments against the Oracle cloud infrastructure, platforms, and applications. These tests are intended to validate and improve the overall security of Oracle Cloud Services.

However, Oracle does not assess or test any components (including, non-Oracle applications, non-Oracle databases or other non-Oracle software, code or data, as may be applicable) that you manage through or introduce into – including introduction through your development in or

creation in - the Oracle Cloud Services (the "**Customer Components**"). This policy does not address or provide any right to conduct testing of any third party materials included in the Customer Components.

Except as otherwise permitted or restricted in your Oracle Cloud Services agreements, your service administrator who has system level access to your Oracle Cloud Services may run penetration and vulnerability tests for the Customer Components included in certain of your Oracle Cloud Services in accordance with the following rules and restrictions.

Permitted Cloud Penetration and Vulnerability Testing

The following explains where penetration and vulnerability testing of Customer Components is permitted:

- **IaaS:** Using your own monitoring and testing tools, you may conduct penetration and vulnerability tests of your acquired single-tenant Oracle Infrastructure as a Service (IaaS) offerings. You must notify Oracle prior to conducting any such penetration and vulnerability tests in accordance with the process set forth below. Pursuant to such penetration and vulnerability tests, you may assess the security of the Customer Components; however, you may not assess any other aspects or components of these Oracle Cloud Services including the facilities, hardware, software, and networks owned or managed by Oracle or its agents and licensors.
- **PaaS:** Using your own monitoring and testing tools, you may conduct penetration and vulnerability tests of your acquired single-tenant PaaS offerings. You must notify Oracle prior to conducting any such penetration and vulnerability tests in accordance with the process set forth below. Pursuant to such penetration and vulnerability tests, you may assess the security of the Customer Components; however, you may not assess any other aspects or components of these Oracle Cloud Services including the facilities, hardware, networks, applications, and software owned or managed by Oracle or its agents and licensors. To be clear, you may not assess any Oracle applications that are installed on top of the PaaS service.
- **SaaS:** Penetration and vulnerability testing is not permitted for Oracle Software as a Service (SaaS) offerings.

Rules of Engagement

The following rules of engagement apply to cloud penetration and vulnerability testing:

- Your testing must not target any other subscription or any other Oracle Cloud customer resources, or any shared infrastructure components.
- You must not conduct any tests that will exceed the bandwidth quota or any other subscribed resource for your subscription.
- You are strictly prohibited from utilizing any tools or services in a manner that perform Denial-of-Service (DoS) attacks or simulations of such, or any "load testing" against any Oracle Cloud asset including yours.
- Any port scanning must be performed in a non-aggressive mode.
- You are responsible for independently validating that the tools or services employed during penetration and vulnerability testing do not perform DoS attacks, or simulations of such, prior to assessment of your instances. This responsibility includes ensuring any contracted third parties perform assessments in a manner that does not violate this policy.
- Social Engineering of Oracle employees and physical penetration and vulnerability testing of Oracle facilities is prohibited.
- You must not attempt to access another customer's environment or data, or to break out of any container (for example, virtual machine).
- Your testing will continue to be subject to terms and conditions of the agreement(s) under which you purchased Oracle Cloud Services, and nothing in this policy shall be deemed to grant you additional rights or privileges with respect to such Cloud Services.
- If you believe you have discovered a potential security issue related to Oracle Cloud, you must report it to Oracle within 24 hours by conveying the relevant information to My Oracle Support. You must create a service request within 24 hours and must not disclose this information publicly or to any third party. Note that some of the vulnerabilities and issues you may discover may be resolved by you by applying the most recent patches in your instances.
- In the event you inadvertently access another customer's data, you must immediately terminate all testing and report it to Oracle within one hour by conveying the relevant information to My Oracle Support.
- You are responsible for any damages to Oracle Cloud or other Oracle Cloud customers

that are caused by your testing activities by failing to abide by these rules of engagement.

Notification Process

The process for notifying Oracle of Your election to conduct a penetration or vulnerability test as required by this policy can be found in Submitting a Cloud Security Testing Notification.

DATA SCRAPING TOOLS

Any use of data scraping tools or technologies with Oracle Cloud Services to collect data available through any Oracle user interface or via web service calls requires the express written permission of Oracle. Oracle reserves the right to require that your proposed data scraping tools are validated and tested by Oracle prior to use in production, and are subsequently re-validated and tested annually.

Security Best Practices

This guide provides actionable guidance and recommendations to Oracle Cloud Infrastructure customers for securely configuring Oracle Cloud Infrastructure services and resources. Understanding Oracle Cloud Infrastructure services and their security features is an essential prerequisite before reading. As a prerequisite, Oracle recommends that you become familiar with security of services. For more information, see Oracle Cloud Infrastructure Security Guide.

Security of an Oracle Cloud Infrastructure tenancy is based on a combination of factors, all of which must be thought through and securely configured. From a practical perspective, take a hierarchical view of Oracle Cloud Infrastructure tenancy security configuration, where we start with addressing the foundational security issues. The following steps provide a roadmap of high-level guidelines to follow when configuring security of a tenancy, where we provide a link to a section enumerating detailed security guidance related to each step.

- **User authentication and authorization:** The initial step in securely configuring a tenancy is to create mechanisms for authenticating users and authorizing users to access tenancy resources in a least-privilege manner. This step comprises creating Oracle Cloud Infrastructure Identity and Access Management (IAM) users, creating IAM

groups, formulating authentication mechanisms (for example, Console access using password, API access using API keys, and auth token for object store) for the IAM users created, grouping customer tenancy resources into logical groups using compartments, and formulating IAM security policies authorizing access of IAM groups to tenancy or compartment resources. For enterprises, federating their on-premises users and groups to their tenancy is an important consideration. IAM allows you to create users, groups, security polices, and federation mechanisms. Security recommendations for configuring IAM are provided in the IAM section.

- **Network security architecture:** After formulating IAM user authentication and authorization, a next step is creating a network security architecture for securely running the customer applications and storing their data in a tenancy. All the customer's compute and storage resources are enclosed in a virtual cloud network (VCN) created for the customer. VCN is a software-defined network, resembling the on-premises physical network used by customers to run their workloads. Formulating a VCN security architecture includes tasks such as:
    - Creating VCN subnets for network segmentation
    - Formulating VCN and load balancer firewalls using VCN security lists
    - Using load balancing for high-availability and TLS
    - Determining type of VCN external connectivity whether internet, on-premises network, peered VCN, or combination of these
    - Using virtual network security appliances (for example, next-generation firewalls, IDs)
    - Creating DNS zones and mappings. An important security consideration in load balancers is using customer Transport Layer Security (TLS) certificates to configure TLS connections to customer's VCN. Security recommendations for VCN are provided in the Networking section.
- **Compute instances security configuration:** Within a customer VCN, the customer applications run on compute instances including Bare Metal (BM) instances, Virtual Machine (VM) instances and GPUs. Compute instances are the basic compute building blocks. Bare metal instances have no Oracle-managed software running on them, with

the result that the instances and data stored (in memory and local drives) are completely controlled by the customer. VM instances are architected with least privilege mechanisms, and with corporate industry-leading hypervisor security best-practices. Depending on their security and performance requirements, customers have a choice of using BM and VM instances, to run their application workloads in their tenancy. It is imperative to securely configure compute instances, to maintain security of customer applications running on them. Recommendations related to instance security configuration with respect to instance firewalls, instance credential management, entropy, security patching, and security logging and monitoring are provided in the Compute section.

- **Data storage security configuration:** Depending on the type of data and access required, customers can store data in local drives (attached to compute instances), remote block volumes, object store buckets, databases, or file storage in their tenancy. To handle these data storage requirements, Oracle Cloud Infrastructure offers multiple data storage services such as Block Volume, Object Storage, Database, and File Storage. In order to meet their data security requirements, customers need to formulate a tenancy data storage architecture for storing their data in their tenancy, and securely configure the storage services used. Compliance and regulatory requirements are an important factor in determining an appropriate data storage security architecture. Recommendations related to storage services security configuration are available in Block Volume, Object Storage, Database, and File Storage sections. In addition to these services, customers need to consider security of data stored ephemerally in compute instance memory (DRAM) and local NVMe storage.

API Audit logs record calls to APIs (for example, through the Console, SDKs, CLIs, and custom clients using the APIs) as log events. The API Audit logs are always on by default and can't be turned off. These logs are available to customers for 90 days, with retention period configurable up to 365 days. Information in the API Audit logs show what time API activity occurred, the source of the activity, the target of the activity, what the action was, and what the response was. Oracle recommends that customers periodically review the OCI API Audit logs to ensure they are in accordance with actions they took on their tenancy resources.

For service-specific best practices, see the following topics:

- Securing Block Volume
- Securing Compute
- Securing Data Transfer
- Securing Database
- Securing Email Delivery
- Securing File Storage
- Securing IAM
- Securing Networking: VCN, Load Balancers, and DNS
- Securing Object Storage
- Securing Resource Manager

Securing Block Volume

SECURITY RECOMMENDATIONS

- There are two types of volumes: block volumes and boot volumes. Block volumes allow instance storage capacity to be expanded dynamically. A boot volume contains the image used to boot the compute instance. The IAM service groups the family of related volume resource types into a combined resource type called `volume-family`.
- Assign least privilege access for IAM users and groups to resource types in `volume-family`. The resource types in `volume-family` are`volumes` ,`volume-attachments`, and `volume-backups`. The`volume-family` resources are detachable block volume devices that allow dynamic expansion of instance storage capacity or contain the image for booting the instance. The `volume-attachments` resources are attachments between volumes and instances. The `volume-backups` resources are point-in-time copies of volumes that can be used to create block volumes or recover block volumes.

DATA DURABILITY

To minimize loss of data due to inadvertent deletes by an authorized user or malicious deletes, Oracle recommends to giving `VOLUME_DELETE`, `VOLUME_ATTACHMENT_DELETE` and

`VOLUME_BACKUP_DELETE` permissions to a minimum possible set of IAM users and groups. `DELETE` permissions should be given only to tenancy and compartment administrators.

To minimize loss of data due to deletes or corruption, Oracle recommends that you make periodic backups of volumes. Oracle Cloud Infrastructure allows automated scheduled backups. For more information about scheduled backups, see Policy-Based Backups.

Data-at-rest Encryption

By default, volumes and their backups are encrypted at rest using AES-256. You can also encrypt your data volumes using tools like dm-crypt, veracrypt, and Bit-Locker. Instructions on dm-crypt encryption are presented in the next section.

Security Policy Examples

Prevent Delete of Volumes

The following example policy allows group `VolumeUsers` to perform all actions on volumes and backups, except deleting them.

```
Allow group VolumeUsers to manage volumes in tenancy
 where request.permission!='VOLUME_DELETE'
Allow group VolumeUsers to manage volume-backups in tenancy
 where request.permission!='VOLUME_BACKUP_DELETE'
```

If `VolumeUsers` can't detach volumes from instances, you can add the following policy to the previous example.

```
Allow group VolumeUsers to manage volume-attachments in tenancy
 where request.permission!='VOLUME_ATTACHMENT_DELETE'
```

Security-related Tasks

Encrypting Non-root Volumes with dm-crypt

dm-crypt is a kernel-level encryption mechanism (part of Linux device mapper framework) to provide encrypted volumes. It encrypts data passed from the filesystem (for example, ext4 and NTFS), and stores it on a storage device in Linux Unified Key Setup (LUKS) format. The encrypted volumes can be stored on a complete disk, disk partition, logical volume, or a file-backed storage created using loopback devices. Cryptsetup is the user-level utility used to

manage dm-crypt, and used to encrypt partitions and files. dm-crypt uses the Linux crypto APIs for encryption routines.

1. Attach block storage volume to an instance (for example, `/dev/sdb`)
2. Format `/dev/sdb` for LUKS encryption. Enter LUKS passphrase when prompted. The passphrase is used to encrypt the LUKS master key used for encrypting the volume.

   ```
   cryptsetup -y luksFormat /dev/sdb
   ```

3. Verify that the LUKS formatting is successful.

   ```
   cryptsetup isLuks /dev/sdb && echo Success
   ```

4. Get encryption information about the device.

   ```
   cryptsetup luksDump /dev/sdb
   ```

5. Get LUKS UUID of the device. The UUID value is used to configure the `/etc/crypttab`.

   ```
   cryptsetup luksUUID /dev/sdb
   ```

6. Create a LUKS container with device name, `dev_name`. This also creates a device node, `/dev/mapper/<dev_name>`.

   ```
   cryptsetup luksOpen /dev/sdb <dev_name>
   ```

7. Get information about the mapped device.

   ```
   dmsetup info <dev_name>
   ```

8. Format the device node as ext4 filesystem.

   ```
   sudo mkfs -t ext4 /dev/sdb
   ```

9. Mount the device node.

   ```
   mount /dev/mapper/<dev_name> /home/encrypt_fs
   ```

10. Add an entry to `/etc/crypttab`.

    ```
    <dev_name> UUID=<LUKS UUID of /dev/sdb> none
    ```

    All the files copied to `/home/encrypt_fs` are encrypted by LUKS.

11. Add a keyfile to an available keyslot of the encrypted volume. This keyfile can be used to access the encrypted volume.

```
dd if=/dev/urandom of=$HOME/keyfile bs=32 count=1
chmod 600 $HOME/keyfile
cryptsetup luksAddKey /dev/sdb ~/keyfile
```

12. Verify the encryption status of files.

```
cryptsetup status /home/encrypt_fs
```

13. Unmount after you're finished.

```
umount /home/encrypt_fs
cryptsetup luksClose <dev_name>
```

To access the encrypted volume:

```
cryptsetup luksOpen /dev/sdb <dev_name> --key-file=/home/opc/keyfile
mount /dev/mapper/<dev_name> /home/encrypt_fs
```

If you lose the keyfile, or if the keyfile or passphrase gets corrupted, you can't decrypt the encrypted volume . This results in permanent loss of data. Oracle recommends that you store durable copies of the keyfile on an on-premises host.

REMOTE MOUNTING OF DM-CRYPT ENCRYPTED DATA VOLUMES

The following steps assume that the keyfile is on an on-premises host (`SRC_IP`) and that `<OCI_SSH_KEY>` is the SSH private key of the instance.

1. Copy keyfile from the on-premises host to an instance.

```
scp -i <OCI_SSH_KEY> keyfile opc@SRC_IP:/home/opc
```

2. Open the encrypted volume.

```
ssh i <OCI_SSH_KEY> opc@SRC_IP "cryptsetup luksOpen /dev/sdb <dev_name> --key-
file=/home/opc/keyfile"
```

3. Mount the volume.

```
ssh -i <OCI_SSH_KEY> opc@SRC_IP "mount /dev/mapper/<dev_name> /home/encrypt_fs"
```

4. Perform operations on data in the mounted volume.
5. Unmount the encrypted volume.

```
ssh -i <OCI_SSH_KEY> opc@SRC_IP "umount /home/encrypt_fs"
ssh -i <OCI_SSH_KEY> opc@SRC_IP "cryptsetup luksClose <dev_name>"
```

6. Delete the keyfile from the instance.

```
ssh -i <OCI_SSH_KEY> opc@SRC_IP "\rm -f /home/opc/keyfile"
```

Securing Compute

Security Recommendations

Oracle Cloud Infrastructure Compute provides both bare metal and virtual machine (VM) instances, architected and managed in accordance with industry-leading security best practices.

Managing Instances and Credentials

- To prevent inadvertent or malicious termination of critical instances (for example, production instances), Oracle recommends that you give `INSTANCE_DELETE` permission to a minimal set of groups. Give `DELETE` permissions only to tenancy and compartment admins.
- Instances can be authorized to access Oracle Cloud Infrastructure services (Compute, Block volume, Networking, Load balancing, Object storage), on behalf of an IAM user, by using Oracle Cloud Infrastructure instance principals feature. To use this feature, you create dynamic groups and grant them access to service APIs. Members of dynamic groups are instances that you define based on rules to match instances to the group. A short-lived private key to sign API calls, is delivered through instance metadata service (http://169.254.169.254/opc/v1/identity/cert.pem), and the key is rotated multiple times a day. For more information about accessing services from instances, see Calling Services from an Instance.

Instance Metadata Access Control

- Instance metadata (http://169.254.169.254) provides predefined instance information (for example, OCID and display name), along with custom fields. Short-lived credentials, such as dynamic group credentials, could be provided through the instance metadata. Oracle recommends that you limit instance metadata access to only

privileged users on the instance. For example, iptables can be used to restrict instance metadata access only to privileged users such as root, using the following rule.

```
iptables -A OUTPUT -m owner ! --uid-owner root -d 169.254.169.254 -j DROP
```

- Instances use link local addresses to access instance metadata service (169.254.169.254:80), DNS (169.254.169.254:53), NTP (169.254.169.254:123), kernel updates (169.254.0.3), and iSCSI connections to boot volumes (169.254.0.2:3260, 169.254.2.0/24:3260). Host-based firewalls such as iptables can be used to authorize only root user to access these IPs. Ensure that these OS firewall rules are not altered.

INSTANCE NETWORK ACCESS CONTROLS

- Harden secure shell (SSH) on all instances. Some SSH security recommendations are shown in the following table. SSH configuration options can be set in the sshd_config file (located at /etc/ssh/sshd_config in Linux).

Security Recommendation	Configuration sshd_ config	Comments
Use public-key logins only	PubkeyAuthentication yes	Periodically review SSH public keys in ~/.ssh/authorized_keys file
Disable password logins	PasswordAuthentication no	Mitigates password brute-force attacks
Disable root logins	PermitRootLogin no	Prevents root privileges for remote logins
Change SSH port to non-standard port	Port <port number>	This is optional. Ensure that this change does not break applications using port 22 for SSH

- Secure SSH private keys used to access instances and prevent any inadvertent disclosure. For more information about creating an SSH key pair and configuring an instance with the keys, see Creating a Key Pair.

- Use VCN security lists as a mechanism to allow instance access from authorized IP addresses. Fail2ban is an application that blacklists IP addresses involved in brute-force login attempts (that is, too many failed attempts to an instance). Fail2ban inspects SSH accesses by default, and can be configured for other protocols. For more information about Fail2ban, see Fail2ban Main Page.
- In addition to VCN security lists, host-based firewalls (such as iptables and firewalld) can be used to restrict network access to instances, in terms of ports, protocols, and packet types. These firewalls can be used to prevent potential network security attack reconnaissance (for example, port scanning) and intrusion attempts. Custom firewall rules can be configured, saved, and initialized on every instance boot. The following example shows commands for iptables.

```
# save iptables rules after configuration
sudo iptables-save > /etc/iptables/iptables.rules
# restore iptables rules on next reboot
sudo /sbin/iptables-restore < /etc/iptables.rules
# restart iptables after restore
sudo service iptables restart
```

INSTANCE ENTROPY

Both bare metal and VM instances provide high-quality and high-throughput entropy source. Instances have hardware random number generator support, whose output is fed into the entropy pools used by the OS to generate random numbers. In Linux instances, `/dev/random` is non-blocking, and recommended to be used for security applications requiring random numbers. You can use the following commands to check the throughput and quality of random numbers generated by `/dev/random`, before using its output in applications.

```
# check sources of entropy
sudo rngd -v
# enable rngd, if not already
sudo systemctl start rngd
# verify rngd status
sudo systemctl status rngd
# verify /dev/random throughput and quality using rngtest
cat /dev/random | rngtest -c 1000
```

HOST SECURITY HARDENING AND PATCHING

- Establish baseline for security hardening of OS images (Linux, Windows) running on instances. For more information about security hardening of Oracle Linux images, see Tips for Hardening an Oracle Linux Server. The Center for Internet Security Benchmarks provides a comprehensive set of OS security hardening benchmarks for various distributions of Linux and Windows Server.
- Keep the instance software up-to-date with security patches. Oracle recommends that you periodically apply the latest available software updates to your instances. For Oracle Linux, you can run `sudo yum update` command. On Oracle Linux, you can get information about available and installed security patches using the `yum-security` plugin. Commands for yum-security are available below. For Oracle Linux instances launched after February 15, 2017, Ksplice support is available for applying patches without rebooting the instance. For more information about using Ksplice on Oracle Cloud Infrastructureinstances, see Installing and Running Oracle Ksplice.

```
# Install yum-security plugin
yum install yum-plugin-security
# Get list of security patches without installing them
yum updateinfo list security all
# Get list of installed security patches
yum updateinfo list security all
```

INSTANCE SECURITY LOGGING AND MONITORING

- Various security-related events are captured in log files. Oracle recommends that you periodically review these log files for detecting any security issues. In Oracle Linux, the log files are located in `/var/log`. Some security-relevant log files are listed in the table below.

Log File or Directory	Description
`/var/log/secure`	Auth log showing failed and successful logins
`/var/log/audit`	Auditd logs capturing system calls issued, sudo attempts, user logins, etc. ausearch and aureport are two tools used to query auditd logs
`/var/log/yum.log`	Lists various packages installed or updated on the instance with yum
`/var/log/cloud-init.log`	cloud-init can run user-provided scripts as privileged user, during instance boot. For example, SSH keys can be introduced using cloud-init. Oracle recommends that you review the cloud-init logs for any unrecognized commands.

- Host-based intrusion detection system (IDS) monitors instances for unauthorized accesses. OSSEC and Wazuh (OSSEC fork) are popular open-source IDS that can monitor for unauthorized access, malware, file modifications, and security misconfigurations. In the case of Wazuh, Wazuh server and ELK stack are deployed on an instance, and agents are deployed on other instances in the VCN to send logs to the Wazuh server. The resulting alerts are displayed on a Kibana dashboard. Wazuh IDS was prototyped on instances, and below are instructions for deploying a working Wazuh server on an instance (with ELK version 5.6.3). For more information about installing Wazuh agents and accessing the Kibana dashboard, see the Wazuh documentation.

```
#!/bin/sh

echo "installing elasticsearch"
sudo add-apt-repository ppa:webupd8team/java;
sudo apt-get update;
sudo apt-get install oracle-java8-installer;

sudo apt-get install curl apt-transport-https
curl -s https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add - ;
echo "deb https://artifacts.elastic.co/packages/5.x/apt stable main" | sudo tee
```

```
/etc/apt/sources.list.d/elastic-5.x.list;
sudo apt-get update;

sudo apt-get install elasticsearch=5.6.3;

sudo systemctl daemon-reload
sudo systemctl enable elasticsearch.service
sudo systemctl start elasticsearch.service

curl https://raw.githubusercontent.com/wazuh/wazuh-kibana-app/2.1/server/startup/integration_
files/template_file.json | curl -XPUT 'http://localhost:9200/_template/wazuh' -H 'Content-Type:
application/json' -d @-

curl https://raw.githubusercontent.com/wazuh/wazuh-kibana-app/2.1/server/startup/integration_
files/alert_sample.json | curl -XPUT "http://localhost:9200/wazuh-a
lerts-"`date +%Y.%m.%d`"/wazuh/sample" -H 'Content-Type: application/json' -d @-

echo "installing logstash"
sudo apt-get install logstash=1:5.6.3-1;
sudo curl -so /etc/logstash/conf.d/01-wazuh.conf
https://raw.githubusercontent.com/wazuh/wazuh/2.1/extensions/logstash/01-wazuh.conf;
sudo curl -so /etc/logstash/wazuh-elastic5-template.json
https://raw.githubusercontent.com/wazuh/wazuh/2.1/extensions/elasticsearch/wazuh-elastic5-
template.json;
sudo usermod -a -G ossec logstash;

sudo systemctl daemon-reload;
sudo systemctl enable logstash.service;
sudo systemctl start logstash.service;

echo "installing kibana"
sudo apt-get install kibana=5.6.3;
sudo -u kibana /usr/share/kibana/bin/kibana-plugin install
https://packages.wazuh.com/wazuhapp/wazuhapp-2.1.1_5.6.3.zip;

sudo systemctl daemon-reload;
sudo systemctl enable kibana.service;
sudo systemctl start kibana.service;
```

SECURITY POLICY EXAMPLES

In all the following examples, the policies are scoped to a tenancy. However, by specifying a compartment name, they can be scoped down to specific compartment in a tenancy.

RESTRICT USERS ABILITY TO DELETE INSTANCES

The following example allows the `InstanceUsers` group to launch instances but not delete them

```
Allow group InstanceUsers to manage instance-family in tenancy
 where request.permission!='INSTANCE_DELETE'
Allow group InstanceUsers to use volume-family in tenancy
Allow group InstanceUsers to use virtual-network-family in tenancy
```

RESTRICT ABILITY TO USE INSTANCE CONSOLE

For security compliance reasons, some customers do not want to expose instance console to users in their tenancy. The following policy example restricts ability to create or read from consoles.

```
Allow group InstanceUsers to manage instance-console-connection in tenancy
 where all {request.permission!= INSTANCE_CONSOLE_CONNECTION_READ,
            request.permission!= INSTANCE_CONSOLE_CONNECTION_CREATE}
```

Securing Data Transfer

Oracle offers offline data transfer solutions that let you migrate large amounts of data to buckets in a tenancy in Oracle Cloud Infrastructure. Data transfer solutions include:

- Data Transfer Disk
  For more information about securely transferring data using this service, see Secure Disk Data Transfer to Oracle Cloud Infrastructure
- Data Transfer Appliance
  For more information about securely transferring data using this service, see Secure Disk Data Transfer to Oracle Cloud Infrastructure.

Securing Database

Security Recommendations

This section lists security recommendations for managing Oracle Cloud Infrastructure Database instances. Recommendations for securely configuring Oracle databases are available in the Oracle Database Security Guide.

Database Access Control

- Users authenticate to the database using their password. Oracle recommends that these passwords be strong. For guidelines on choosing Oracle database passwords, see Guidelines for Securing Passwords. In addition, Oracle database provides a PL/SQL script to verify database password complexity. This script is located at `$ORACLE_HOME/rdbms/admin/UTLPWDMG.SQL`. For instructions on running UTLPWDMG.SQL script to verify password complexity, see Enforcing Password Complexity Verification.
- In addition to the database password, you can use VCN security lists to enforce network access control to database instances. Oracle recommends that you configure VCN security lists to allow least privilege access to customer databases in Oracle Cloud Infrastructure Database.
- DB systems created within a public subnet can send outbound traffic directly to the Internet. DB systems created within a private subnet do not have internet connectivity, and internet traffic (both egress and ingress) cannot reach the instance directly. If you try to define a route to a DB system within a private subnet using an internet gateway, the route is ignored.

  To perform OS patching and backup for a DB system on private subnet, you can use a service gateway or a NAT gateway to connect to your patching or backup endpoints.

  In an virtual cloud network (VCN), you can use security rules along with a private subnet to restrict access to a DB system. In multi-tier deployments, a private subnet and VCN security rules can be used to restrict access to the DB system from the application tiers.

DATA DURABILITY

- Oracle recommends that you give database delete permissions (`DATABASE_DELETE`, `DB_SYSTEM_DELETE`) to a minimum possible set of IAM users and groups. This minimizes loss of data due to inadvertent deletes by an authorized user or due to malicious deletes. Only give `DELETE` permissions to tenancy and compartment administrators.
- You can use RMAN to do periodic backups of Database databases, where encrypted backup copies are stored in local storage (block volumes, for example) or Oracle Cloud Infrastructure Object Storage. RMAN encrypts each backup of a database with a unique encryption key. In transparent mode, the encryption key is stored in the Oracle Wallet. RMAN backups to Object Storage require internet gateway (IGW), and VCN security lists need to be configured to allow secure access to Object Storage. For information about configuring VCN security lists for backing up bare metal databases, see Backing Up to Oracle Cloud Infrastructure Object Storage. For information about backing up and Exadata databases, see Backing Up an Exadata Database.

DATABASE ENCRYPTION AND KEY MANAGEMENT

- All databases created in Oracle Cloud Infrastructure are encrypted using transparent data encryption (TDE). Note that if you migrate an unencrypted database from on-premise to Oracle Cloud Infrastructure using RMAN, the migrated database will not be encrypted. Oracle strongly recommends encrypting such databases after migrating them to the cloud.

  To learn how to encrypt your database with minimum downtime during migration, see the Oracle Maximum Availability Architecture white paper Converting to Transparent Data Encryption with Oracle Data Guard using Fast Offline Conversion.

  Note that virtual machine DB systems use Oracle Cloud Infrastructure block storage instead of local storage. Block storage is encrypted by default.
- User-created tablespaces are encrypted by default in Oracle Cloud Infrastructure Database. In these databases, `ENCRYPT_NEW_TABLESPACES` parameter is set to `CLOUD_ONLY` where tablespaces created in a Database Cloud Service (DBCS) database are transparently encrypted with the AES128 algorithm unless a different algorithm is specified.

- The Database administrator creates a local Oracle Wallet on a newly created database instance, and initializes the Transparent Data Encryption (TDE) master key. Then the Oracle Wallet is configured to be "auto-open". However, a customer can choose to set a password for the Oracle Wallet, and Oracle recommends that you set a strong password (eight characters or more, with at least one capital letter, one small letter, one number, and one special symbol).
- Oracle recommends that you periodically rotate the TDE master key. The recommended rotation period is 90 days or less. You can rotate the TDE master key by using native database commands ("administer key management" in 12c, for example) or dbaascli. All previous versions of TDE master key are maintained in the Oracle Wallet.
- Oracle Key Vault (OKV) is a key management appliance used for managing Oracle TDE master keys. OKV can store, rotate, and audit accesses to TDE master keys. For instructions about installing and configuring OKV in Oracle Cloud Infrastructure, see Managing Oracle Database Encryption Keys in Oracle Cloud Infrastructure with Oracle Key Vault.

DATABASE PATCHING

Applying Oracle database security patches (Oracle Critical Patch Updates) is imperative to mitigate known security issues, and Oracle recommends that you keep patches up-to-date. Patchsets and Patch Set Updates (PSUs) are released on a quarterly basis . These patch releases contain security fixes and additional high-impact/low-risk critical bug fixes.

For information about the latest known security issues and available fixes, see Critical Patch Updates, Security Alerts and Bulletins. If your application does not support the latest patches and needs to use a DB system with older patches, you can provision a DB system with an older version of the Oracle Database edition you are using. In addition to reviewing the critical patch updates and security alerts for your Oracle Database, Oracle recommends that you analyze and patch the operating system provisioned with the DB system.

For information about applying patches to Oracle Cloud Infrastructure Database instances, see Patching a DB System and Patching an Exadata DB System.

Database Security Configuration Checking

- The Oracle Database Security Assessment Tool (DBSAT) provides automated security configuration checks of Oracle databases in Oracle Cloud Infrastructure. DBSAT performs security checks for user privilege analysis, database authorization controls, auditing polices, database listener configuration, OS file permissions, and sensitive data stored. Oracle database images in Oracle Cloud Infrastructure Database are scanned with DBSAT before provisioning. After provisioning, Oracle recommends that you periodically scan databases with DBSAT, and remediate any issues found. DBSAT is available free of charge to Oracle customers.

Database Security Auditing

Oracle Audit Vault and Database Firewall (AVDF) monitors database audit logs and creates alerts. For instructions about installing and configuring AVDF in Oracle Cloud Infrastructure, see Deploying Oracle Audit Vault and Database Firewall in Oracle Cloud Infrastructure.

Database Backups

Oracle recommends using Managed backups (backups created using the Oracle Cloud Infrastructure Console or the API) whenever possible. When you use managed backups, Oracle manages the object store user and credentials, and rotates these credentials every 3 days. Oracle Cloud Infrastructure encrypts all managed backups in the object store. Oracle uses the Database Transparent Encryption feature by default for encrypting the backups.

If you are not using managed backups, Oracle recommends that you change the object store passwords at regular intervals.

Security Policy Examples

Prevent Delete of Database Instances

The following example policy allows the group `DBUsers` to perform all management actions except delete databases and any artifacts.

```
Allow group DBUsers to manage db-systems in tenancy
 where request.permission!='DB_SYSTEM_DELETE'
Allow group DBUsers to manage databases in tenancy
 where request.permission!='DATABASE_DELETE'
```

```
Allow group DBUsers to manage db-homes in tenancy
 where request.permission!='DB_HOME_DELETE'
```

Securing Email Delivery

The Email Delivery service offers an SMTP endpoint, secured by a password generated in the Console. The SMTP password is required for sending emails using Email Delivery. Oracle recommends that you create a separate IAM user for SMTP. This user must have manage permissions for `approved-senders` and `suppressions` resource types. Oracle recommends that you securely store the SMTP credential, and periodically rotate it. For more information about generating an SMTP credential for Email Delivery, see Generate SMTP Credentials for a User.

For Email Delivery best practices, including managing your sender reputation and help for avoiding being blacklisted, see Deliverability Best Practices.

Securing File Storage

The File Storage Service exposes an NFSv3 endpoint as a mount target in each customer's VCN subnet. The mount target is identified by a DNS name and is mapped to an IP address. Oracle recommends that you use VCN security lists (of the mount target subnet) to configure network access to the mount target from only authorized IP addresses.

You can mount a file system using the Console or from a Linux command line using NFS utilities. You can authorize users to mount file systems using IAM security policies, but this applies to the console only.

For data durability, Oracle recommends that you take periodic snapshots of the file system. To minimize accidental deletion of data, constrain the set of users having privileges to delete mount targets, file-systems, and snapshots.

All file-system data is encrypted at rest using AES-128.

Access to mounted NFS file systems from a remote host is determined by POSIX user and group permissions. Oracle recommends that you use well-known NFS security best practices such as the `all_squash` option to map all users to `nfsnobody`, and NFS ACLs to enforce access control to the mounted file system.

SECURITY POLICY EXAMPLES

PREVENT MOUNT TARGET AND FILE SYSTEM DELETION

The following example prevents group FileUsers from deleting mount targets and file-systems.

```
Allow group FileUsers to manage file-systems in tenancy
 where request.permission!='FILE_SYSTEM_DELETE'
Allow group FileUsers to manage mount-targets in tenancy
 where request.permission!='MOUNT_TARGET_DELETE'
Allow group FileUsers to manage export-sets in tenancy
 where request.permission!='EXPORT_SET_DELETE'
```

Securing IAM

SECURITY RECOMMENDATIONS

Oracle Cloud Infrastructure Identity and Access Management (IAM) provides authentication of users, and authorization to access resources. Security-relevant IAM concepts include:

IAM concepts and descriptions

Concept	Description
Compartment	A compartment is a fundamental mechanism to aggregate resources into logical groups. They also provide isolation.
Tenancy	Oracle Cloud Infrastructure automatically creates a tenancy for an account created by an organization. It is the root compartment that contains all the organization's resources.
Users and groups	A group is an aggregation of users who need similar access to a group of resources.
Resource	A resource is an object created in Oracle Cloud Infrastructure services.

Concept	Description
Security policy	A security policy specifies the type of access IAM groups have to resources in a specified aggregation level. An aggregation level can be the tenancy, a compartment, or a service.
Dynamic groups	A dynamic group allows aggregating Compute instances as principal actors (similar to user groups), in order to authorize instances to make calls to Oracle Cloud Infrastructure APIs.
Tags	Tags allow you to organize resources across multiple compartments for reporting purposes or for taking bulk actions.
Federation	Mechanism to federate IAM with other identity providers (IdP) used by an organization to authenticate their users.

Oracle recommends that you periodically monitor Audit logs to review changes to IAM users, groups, and security policies.

IAM Tenancy and Compartments

- Compartments are unique to IAM, and offer a mechanism that allows an enterprise customer to meet its central needs by having a single account or tenancy. This single account or tenancy provides full central control and visibility while also allowing the account or tenancy to be subdivided to meet the needs of constituent teams, projects, and initiatives.
- For security and governance reasons, users should only have access to resources they need. For example, enterprise users working on a project or belonging to a business unit should have access only to resources belonging to the project or business unit. Compartments provide an effective mechanism to group tenancy resources based on their access privileges and authorize groups of users to access the compartments on as needed basis. In the example above, a compartment can be created to include all resources belonging to a business unit, and authorize only members of the business unit to access the compartment. Similarly, a groups' access to a compartment can be revoked when they do not need it anymore.

- Keep the following in mind when you create a compartment and assign resources:
    - Every resource should belong to a compartment.
    - A resource can't be reassigned to a different compartment after creation.
    - A compartment can't be deleted after creation.
- Resource tags provide a way to logically aggregate resources distributed across multiple compartments. For example, tenancy resources can be tagged as `test` or `production` depending on their use. For more information about resource tags (free-form and defined tags), see Resource Tags.
- Every tenancy comes with a default administrators group. This group can perform any action on all resources in a tenancy (that is, they have root access to the tenancy). Oracle recommends that you keep the group of tenancy administrators as small as possible. Some security recommendations on managing tenancy administrators:
    - Have security policies granting membership of tenancy administrator group strictly on a as-needed basis.
    - Tenancy administrators should use high-complexity passwords, along with MFA, and periodically rotate their passwords.
    - After account set up and configuration, Oracle recommends that you don't use the tenancy administrator account for day-to-day operations. Instead, create less privileged users and groups.
    - Though administrator accounts are not used for daily operations, they are still needed to address emergency scenarios impacting customer tenancy and operations. Specify secure and auditable "break-glass" procedures for using administrator accounts in such emergencies.
    - Disable tenancy administration access immediately when an employee leaves the organization.
    - Because the tenancy administrator group membership is restricted, Oracle recommends that you create security policies which prevent administrator account lock-out (for example, if the tenancy administrator leaves the company and no current employees have administrator privileges).

IAM Users and Groups

- Create an IAM user for everyone in the customer organization who needs access to resources. Do not share IAM user accounts across multiple users, especially those with administrative accounts. Using distinct IAM users enables enforcing least privilege access for each user, and captures their actions in audit logs.
- The recommended unit of administration is IAM groups, which makes it easier to manage and keep track of security permissions (as opposed to individual users). Create IAM groups with permissions to do commonly needed tasks (for example, network administration, volume administration), and assign users to these groups on an as-needed basis. IAM permissions can be used to give a group access to resources across multiple compartments in a tenancy.
- Periodically review membership of IAM users in IAM groups, and remove IAM users from groups they do not need access to anymore. Using group membership to manage user access scales well with increasing number of users.
- Deactivate IAM users who do not need access to tenancy resources. Deleting an IAM user removes the user permanently. You can temporarily deactivate an IAM user by doing the following:
    - Rotate the user password and throw it away.
    - Remove all tenancy permissions of the user by removing membership from all groups.

IAM Credentials

IAM user credentials (Console password, API signing key, auth tokens, and customer secret keys) grant access to resources. It is important to secure these credentials to prevent unauthorized access to Oracle Cloud Infrastructure resources. General guidelines for handling credentials include:

- Create a strong console password for each IAM user, with sufficient complexity. Oracle recommends the following for a complex password:
    - Password has a minimum length of 12 characters
    - Password contains at least one uppercase letter

    - Password contains at least one lowercase letter
    - Password contains at least one symbol
    - Password contains at least one number
- Rotate IAM passwords and API keys regularly, every 90 days or less. In addition to a security engineering best practice, this is also a compliance requirement. For example, PCI-DSS Section 3.6.4 states, "Verify that key-management procedures include a defined cryptoperiod for each key type in use and define a process for key changes at the end of the defined crypto period(s)."
- Do not hard code sensitive IAM credentials directly in software or documents accessible to a wide audience. Examples include code uploaded to GitHub, presentations, or documents available on the internet. There have been known, highly publicized cases of hackers breaching customer cloud accounts, using credentials inadvertently disclosed on public sites. When software applications need to access Oracle Cloud Infrastructure resources, Oracle recommends that you use instance principals. If it is not feasible to use instance principals, other recommendations include using user environment variables to store credentials, and using locally stored credential files with API keys to be used by the Oracle Cloud Infrastructure SDK or CLI.
- Do not share IAM credentials between multiple users.
- By federating the Console login through Oracle Identity Cloud Service, customers can use multifactor authentication (MFA) for IAM users, especially administrators.

When rotating API keys, verify that the rotated keys work as expected before disabling older keys. For information about generating and uploading IAM API keys, see Required Keys and OCIDs. The high-level steps in rotating an API key are:

1. Generate and upload a new API key.
2. Update the SDK and CLI configuration files with the new API key.
3. Verify that the SDK and CLI calls are working correctly with the new key.
4. Disable the old API key. Use ListApiKeys to list all active API keys.

IAM Security Policies

IAM policies are used to govern access of IAM groups to resources in compartments and in the tenancy. Oracle recommends that you assign least privilege access to IAM groups for accessing resources. The common format for IAM policies is shown in the following example.

```
Allow group <group_name> to <verb> <resource-type> in compartment <compartment_name>
Allow group <group_name> to <verb> <resource-type> in tenancy
```

IAM policies allow four predefined verbs: inspect, read, use and manage. Inspect allows least privilege and manage allows the maximum. The four verbs are shown in increasing order of privilege in the following table.

IAM policy verbs

Verb	Access Type	Example User
`inspect`	Should only show metadata. This usually results in ability to list resources only	third-party auditor
`read`	inspect plus ability to read resource and user metadata. This is the permission most users need to get work done.	internal auditors
`use`	read plus ability to work with resources (the actions vary by resource type). Excludes ability to create or delete resource	regular users (software developers, system engineers, dev managers, etc) setting up and configuring tenancy resources, and applications running on them
`manage`	All the permissions for all the resources	administrators, executives (for break-glass scenarios)

The resource types of Oracle Cloud Infrastructure resources are shown in the following table.

IAM resource families, descriptions, and resource types

Resource Type Family	Description	Resource Types
`all-resources`	All resource types	
No name by design	Resource types in IAM service	`compartments`, `users`, `groups`, `dynamic-groups`, `policies`, `identity-providers`, `tenancy tag-namespaces`, `tag-definitions`
`instance-family`	Resource types in compute service	`console-histories`, `instance-console-connection`, `instance-images`, `instances`, `volume-attachments`
`volume-family`	Resource types in block storage service	`volumes`, `volume-attachments`, `volume-backups`
`virtual-network-family`	Resource types in virtual networking service	`vcns`, `subnets`, `route-tables`, `security-lists`, `dhcp-options`, `private-ips`, `public-ips`, `internet-gateways`, `local-peering-gatewaysdrgs`, `deg-attachments`, `cpes`, `ipsec-connections`, `cross-connects`, `cross-connect-groups`, `virtual-circuits`, `vnics`, `vnic-attachments`
`object-family`	Resource types in object storage service	`buckets`, `objects`

Resource Type Family	Description	Resource Types
`database-family`	Resource types in DbaaS service	`db-systems`, `db-nodes`, `db-homes`, `databases`, `backups`
`load-balancers`	Resources in Load Balancer service	`load-balancers`
`file-family`	Resources in file storage service	`file-systems`, `mount-targets`, `export-sets`
`dns`	Resources in DNS service	`dns-zones`, `dns-records`, `dns-traffic`
`email-family`	Resources in email delivery service	`approved-senders`, `suppressions`

For more information about IAM verbs and resource type permission mappings, see Details for the Core Services.

IAM security policies can be made fine-grained through conditions. Access specified in the policy is allowed only if the condition statements evaluate to true. Conditions are specified using predefined variables. The variables use the key words `request` or `target`, depending on whether the variable is relevant to the request or the resource being acted on, respectively. For information about supported predefined variables, see Policy Reference.

IAM dynamic groups are used to authorize Compute instances to access Oracle Cloud Infrastructure APIs. The instance principals feature can be used by applications, running on the instances, to programmatically access Oracle Cloud Infrastructure services. Customers create dynamic groups, which include instances as members, and authorize access to their tenancy resources using IAM security policies. All access by instances is captured in the audit logs available to customers.

IAM Federation

- Oracle recommends that you use federation to manage logins into the Console. Identity federation supports SAML 2.0 compliant identity providers, and can be used to federate on-premises users and groups to IAM users and groups. The enterprise administrator needs to set up a federation trust between the on-premises identity provider (IdP) and IAM, in addition to creating mapping between on-premises groups and IAM groups. Then, on-premises users can single sign-on (SSO) into the Console, and access resources based on authorization of IAM groups they belong to. For more information about federating to the Console, see Federating with Identity Providers. Federation is especially important for enterprises using custom policies for user authentication (for example, multifactor authentication) . For more information about managing users and groups under federation, see Federating with Identity Providers.
- When using federation, Oracle recommends that you create a federation administrators group that maps to the federated IdP administrator group. The federation administrators group will have administrative privileges to manage customer tenancy, while being governed by the same security policies as the federated IdP administrator group. In this scenario, it is a good idea to have access to the local tenancy administrator user (that is, member of the default tenancy administrator IAM group), to handle any break-glass type scenarios (for example, inability to access resources through federation). However, you must prevent any unauthorized use of this highly privileged local tenancy administrator user. Oracle recommends the following approach to securely managing the tenancy administrator user:
    1. Create a local user belonging to the default tenancy administrator group.
    2. Create a highly complex Console password or passphrase (18 characters or more, with at least one lowercase letter, one uppercase letter, one number, and one special character) for the local tenancy administrator user.

3. Securely escrow the local tenancy administrator user password in an on-premises location (for example, place the password in a sealed envelope in an on-premises physical safe).
4. Create security policies for accessing the escrowed password only under specific "break-glass" scenarios.
5. Have IAM security policy to prevent the federated administrators IAM group from adding or modifying membership of the default tenancy administrator group to prevent security by-passes.
6. Monitor audit logs for accesses by default tenancy administrator and changes to the administrator group, to alert on any unauthorized actions. For additional security, the local tenancy administrator user password can be rotated after every login, or periodically, based on a password policy.

For an example that shows the way various IAM components fit together, see Example Scenario. Periodically monitor Audit logs to review changes to IAM users, groups, policies, compartments, and tags.

Security Policy Examples

Common IAM security policy examples are available at Common Policies. In all the examples that follow, the policies are scoped to a tenancy. However, by specifying a compartment name, you can scope down the policies to specific compartments in a tenancy.

Create Service-level Admins for Least Privilege

To implement security principle of least privilege, you can create service-level admins in the tenancy to further scope down administrative access. This means that service-level administrators can only manage resources of a specific service. For instance, network administrators need administrative (`manage`) access only to VCN resources, and not to other resources. The following example shows how to create administrator groups for block storage (`VolumeAdmins`), VCN (`NetworkAdmins`), databases (`DBAdmins`), and object storage (`StorageAdmins`).

```
Allow group TenancyAdmins to manage all-resources in tenancy
Allow group VolumeAdmins to manage volume-family in tenancy
Allow group NetworkAdmins to manage virtual-network-family in tenancy
```

```
Allow group StorageAdmins to manage object-family in tenancy
Allow group DBAdmins to manage database-family in tenancy
```

You can further constrain the security policies to a specific compartment. For example, the HR department in an enterprise can create group `HRAdmins` to manage resources within its compartment, `HR-compartment`. The `HRNetworkAdmins` group has administrative access to VCN resources only within the `HR-compartment` compartment.

```
Allow group HRAdmins to manage all-resources in compartment HR-compartment
Allow group HRNetworkAdmins to manage virtual-network-family in compartment HR-compartment
```

Compliance auditors are tasked with examining cloud resources and verifying for policy violations. The following policy allows group `InternalAuditors` to inspect (`list`) all resources in a tenancy.

```
Allow group InternalAuditors to inspect all-resources in tenancy
```

If you want to limit auditors to only inspect users and groups in a tenancy, you can create a group `UserAuditors` with the following policy:

```
Allow group UserAuditors to inspect users in tenancy
Allow group UserAuditors to inspect groups in tenancy
```

If you want to create an auditor group that can only inspect VCN firewalls in the tenancy, use the following policy:

```
Allow group FirewallAuditors to inspect security-lists in tenancy
```

In all the policy examples, you can constrain the policies to a compartment by specifying `Compartment` *`<name>`* (where ***<name>*** is the compartment name) in the policy.

Restrict Ability to Change Tenancy Administrators Group Membership

Members in the group `Administrators` can manage all resources in a tenancy. Membership of the `Administrators` group is controlled by users in the group. Usually, it's convenient to have a group to create and add users in the tenancy, but restrict them from making changes to the `Administrators` group membership. The following example creates a group `UserAdmins` to do this.

```
Allow group UserAdmins to inspect users in tenancy
Allow group UserAdmins to inspect groups in tenancy
Allow group UserAdmins to use users in tenancy
 where target.group.name!='Administrators'
```

```
Allow group UserAdmins to use groups in tenancy
 where target.group.name!='Administrators'
```

Use verb with conditions (third and fourth policy statements) allows `UserAdmins` to add users and groups using APIs (`UpdateUser`, `UpdateGroup`) to all groups in the tenancy except the `Administrators` group. However, because `target.group.name!='Administrators'` is not related to the `list` and `get` APIs (`ListUsers`, `GetUser`, `ListGroups`, and `GetGroup`), these APIs will fail. So you must explicitly add the `inspect` verb (first and second policy statements) to allow `UserAdmins` to get user and group membership information.

Prevent Delete or Update of Security Policies

The following example creates a group `PolicyAdmins` to be able to create and list security policies created by tenancy administrators, but not delete or update them.

```
Allow group PolicyAdmins to use policies in tenancy
Allow group PolicyAdmins to manage policies in tenancy
 where request.permission='POLICY_CREATE'
```

This security policy statement explicitly only allows `POLICY_CREATE` permission, and not to `POLICY_DELETE` and `POLICY_UPDATE`.

Prevent Admins from Accessing or Altering User Credentials

Some compliance requirements need separation of duties, especially where user credential management functionality is separated from tenancy management. In this case, you can create two administration groups, `TenancyAdmins` and `CredentialAdmins` where `TenancyAdmins` can perform all tenancy management functions except user credential management, and `CredentialAdmins` can manage user credentials. `TenancyAdmins` can access all APIs except those that list, update, or delete user credentials. `CredentialAdmins` can only manage the user credentials.

```
Allow group TenancyAdmins to manage all resources in tenancy
 where all {request.operation!='ListApiKeys',
            request.operation!='ListAuthTokens',
            request.operation!='ListCustomerSecretKeys',
            request.operation!='UploadApiKey',
            request.operation!='DeleteApiKey',
            request.operation!='UpdateAuthToken',
            request.operation!='CreateAuthToken',
            request.operation!='DeleteAuthToken',
```

```
            request.operation!='CreateSecretKey',
            request.operation!='UpdateCustomerSecretKey',
            request.operation!='DeleteCustomerSecretKey'}
Allow group CredentialAdmins to manage users in tenancy
 where any {request.operation='ListApiKeys',
            request.operation='ListAuthTokens',
            request.operation='ListCustomerSecretKeys',
            request.operation='UploadApiKey',
            request.operation='DeleteApiKey',
            request.operation='UpdateAuthToken',
            request.operation='CreateAuthToken',
            request.operation='DeleteAuthToken',
            request.operation='CreateSecretKey',
            request.operation='UpdateCustomerSecretKey',
            request.operation='DeleteCustomerSecretKey'}
```

USEFUL CLI COMMANDS

In all the following examples, environment variables `$T` and `$C`are set to tenancy OCID and compartment OCID, respectively.

LIST COMPARTMENTS IN A TENANCY

```
# list all compartments (OCID, display name, description) in tenancy $T
oci iam compartment list -c $T
# grep above command for important fields
oci iam compartment list -c $T | grep -E "name|description|\"id\""
```

LIST IAM USERS

```
# lists all users (OCID, display name, description) in tenancy $T
oci iam user list -c $T
# grep above command for important fields
oci iam user list -c $T | grep -E "name|description|\"id\""
```

LIST IAM GROUPS

```
# lists all groups (OCID, display name, description) in tenancy $T.
oci iam group list -c $T
# grep above command for important fields
oci iam group list -c $T | grep -E "name|description|\"id\""
```

LIST USERS IN A GROUP

The following command is helpful for listing users in groups, especially users with administrative privileges. This command requires the OCID of the group whose users are listed.

```
# list users in group with OCID <GROUP_OCID>
oci iam group list-users -c $T --group-id <GROUP_OCID>
```

LIST SECURITY POLICIES

```
# lists all policies (OCID, name, statements) in tenancy $T. Remove pipe to grep to get entire
information
oci iam policy list -c $T
# grep above command for important fields
oci iam policy list -c $T | grep -E "name|Allow|\"id\""
```

Securing Networking: VCN, Load Balancers, and DNS

SECURITY RECOMMENDATIONS

The Networking service has a collection of features for enforcing network access control and securing VCN traffic. These features are listed in the following table.

VCN Feature	Security Description
Public and private subnets	Your VCN can be partitioned into subnets. Subnets have historically been specific to an availability domain, but can now be regional (covering all availability domains in the region). Instances inside private subnets cannot have public IP addresses. Instances inside public subnets can optionally have public IP addresses at your discretion.
Security lists	Security lists provide stateful and stateless firewall capability to control network access to your instances. A security list is configured at the subnet level and enforced at the instance level. You can apply multiple security lists to a subnet. A network packet is allowed if it matches any rule in the security lists.

VCN Feature	Security Description
Gateways	Gateways let resources in a VCN communicate with destinations outside the VCN. The gateways include: • Internet gateway: for internet connectivity (for resources with public IP addresses in public subnets) • NAT gateway: for internet connectivity without exposing the resources to incoming internet connections (for resources in private subnets) • Dynamic routing gateway (DRG): for connectivity to networks outside the VCN's region (for example, your on-premises network by way of an IPSec VPN or FastConnect, or a peered VCN in another region) • Service gateway: for private connectivity to Oracle services such as Object Storage • Local peering gateway (LPG): for connectivity to a peered VCN in the same region
Route table rules	Route tables control how traffic is routed from your VCN's subnets to destinations outside the VCN. Routing targets can be VCN gateways or a private IP address in the VCN.
IAM polices for virtual-network-family	IAM policies specify access and actions permitted by IAM groups to resources in a VCN. For example, IAM polices can give administrative privileges to network administrators who manage the VCNs, and scoped-down permissions to normal users.

Oracle recommends that you periodically monitor Oracle Cloud Infrastructure Audit logs to review changes to VCN security lists, route table rules, and VCN gateways.

NETWORK SEGMENTATION: VCN SUBNETS

- Formulate a tiered subnet strategy for the VCN, to control network access. A common design pattern is to have the following subnet tiers:
  1. **DMZ subnet** for load balancers
  2. **Public subnet** for externally accessible hosts such as NAT instances, intrusion detection (IDS) instances, and web application servers
  3. **Private subnet** for internal hosts such as databases

  No special routing is required for the instances in the different subnets to communicate. However, you can control the types of traffic between the subnets by using the VCN's security lists.
- Instances in the private subnet only have private IP addresses and can be reached only by other instances in the VCN. Oracle recommends that you place security-sensitive hosts (DB systems, for example) in a private subnet, and use security list rules to control the type of connectivity to hosts in a public subnet. In addition to VCN security lists, configure host-based firewalls such as iptables, firewalld for network access control, as a defense-in-depth mechanism.
- You can add a service gateway to your VCN to enable DB systems in the private subnet to directly back up to Object Storage without the traffic traversing the internet. You must set up the subnet's routing and security lists to enable that traffic. For more information, see Access to Oracle Services: Service Gateway.

NETWORK ACCESS CONTROL: VCN SECURITY LISTS

- Use your VCN's security lists to restrict network access to instances in a subnet. A security list is stateful by default, but can also be configured to be stateless. A common practice is to use stateless rules for high-performance applications. In a case where network traffic matches both stateful and stateless security lists, the stateless rule takes precedence. For more information about configuring VCN security lists, see Security Lists.
- To prevent unauthorized access or attacks on Compute instances, Oracle recommends that you use a VCN security list to allow SSH or RDP access only from authorized CIDR blocks rather than leave them open to the internet (0.0.0.0/0). For additional security,

you can temporarily enable SSH (port 22) or RDP (port 3389) access on an as-needed basis using the VCN API UpdateSecurityList. For performing instance health checks, Oracle recommends that you configure VCN security lists to allow ICMP pings. For more information about enabling RDP access, see To enable RDP access in Creating an Instance.

- Oracle recommends bastion hosts as a way to control external access (for example, SSH) to VCN hosts. Usually bastion hosts in a VCN public subnet control access to VCN private subnet hosts. For more information about setting up an SSH bastion host in a VCN, see the white paper Bastion Hosts: Protected Access for Virtual Cloud Networks.
- VCN security lists enable security-critical network access control to Compute instances, and it is important to prevent any unintended or unauthorized changes to security lists. To prevent unauthorized VCN security lists changes, Oracle recommends that you use IAM policies to allow only network administrators to make security list changes.

SECURE CONNECTIVITY: VCN GATEWAYS AND FASTCONNECT PEERING

- VCN gateways provide external connectivity (internet, on-premises, or peered VCN) to VCN hosts. See the table earlier in this topic for a list of the type of gateways. Oracle recommends that you use an IAM policy to allow only network administrators to create or modify VCN gateways.
- Carefully consider allowing internet access to any instances. For example, you don't want to accidentally allow internet access to sensitive database instances. In order for an instance in a VCN to be *publicly accessible from the internet*, you must configure the following VCN options:
    - The instance must be in a VCN public subnet.
    - The VCN containing the instance must have an internet gateway enabled and configured to be the routing target for outbound traffic.
    - The instance must have a public IP address assigned to it.
    - The VCN security list for the public subnet that contains the instance must be configured to `0.0.0.0/0`.

- VPN IPSec provides connectivity between a customer's on-premises network and VCN. You can create two IPSec tunnels for high availability. For more information about creating VPN tunnels to connect VCN DRG to customer CPEs, see VPN Connect.
- FastConnect peering allows you to connect your on-premises network to your VCN with a private circuit so that the traffic does not traverse the public internet. You can set up private peering (to connect to private IP addresses), or public peering (to connect to Oracle Cloud Infrastructure public endpoints, such as for Object Storage). For more information about FastConnect peering options, see FastConnect.

VIRTUAL SECURITY APPLIANCES IN A VCN

- The Networking service lets you implement network security functions such as intrusion detection, application-level firewalls, and NAT (although you can instead use a NAT gateway with your VCN). You can do this by routing all the subnet traffic to a network security host, using route table rules that use a local VCN private IP address as a target. For more information, see Using a Private IP as a Route Target. For high availability, you can assign the gateway security host a secondary private IP address, which you can move to a VNIC on a standby host in case of primary host failure. For more information about setting up a NAT instance in a VCN, see the white paper NAT Instance Configuration: Enabling Internet Access for Private Subnets. Full network packet capture or network flow logs can be captured on the NAT instances using tcpdump, and the logs can be uploaded periodically to an Object Storage bucket.
- Virtual security appliances can be run as virtual machines (VMs) on a bring-your-own-hypervisor (BYOH) model on a bare metal instance. Virtual security appliance VMs running on the BYOH bare metal instance each have their own secondary VNIC, giving direct connectivity to other instances and services in the VNIC's VCN. For information about enabling BYOH on a bare metal instance using an open-source KVM hypervisor, see Installing and Configuring KVM on Bare Metal Instances with Multi-VNIC.
- Virtual security appliances can also be installed on Compute virtual machines (VMs) where VMDK or QCOW2 images of security appliances can be imported using the bring your own image (BYOI) feature. However, due to infrastructure dependencies, the BYOI feature might not work for some appliances, in which case the BYOH model would be another option to use. For more information about importing appliance images into Oracle Cloud Infrastructure, see Bring Your Own Image (BYOI).

Load Balancers

- Oracle Cloud Infrastructure load balancers enable end-to-end TLS connections between a client's applications and a customer's VCN. The TLS connection can be terminated at an HTTP load balancer, or on a back-end server by using a TCP load balancer. The load balancers use TLS1.2 by default. For information about configuring an HTTPS listener, see Managing Load Balancer Listeners. You can also upload your own TLS certificates. For more information see Managing SSL Certificates.
- You can configure network access to load balancers by using VCN security lists. This method provides similar functionality to traditional load balancer firewalls. For public load balancers, Oracle recommends that you use a regional public subnet (for example, DMZ subnet) for instantiating the load balancers in a highly available configuration across two different availability domains. You can configure the load balancer firewall rules by setting up the VCN security lists for that subnet. For more information about creating load balancer security lists, see Update Load Balancer Security Lists and Allow Internet Traffic to the Listener. Similarly, you must configure the VCN security lists for the backend servers to limit traffic only from the public load balancers. For more information about configuring backend server security lists, see Update Rules to Limit Traffic to Backend Servers.

DNS Zones and Records

DNS zones and records are critical for accessibility of web properties. Incorrect updates or unauthorized deletions could result in outage of services, accessed through the DNS names. Oracle recommends that you limit IAM users who can modify DNS zones and records.

Security Policy Examples

Allow Users to Only View Security Lists

Your network administrators are the personnel who should have the ability to create and manage security lists.

However, you may have network users who need to know what rules are in a security list. The following example policy allows the NetworkUsers group to view security lists and their contents. This policy does not let the group create, attach, delete, or modify security lists.

```
Allow group NetworkUsers to inspect security-lists in tenancy
```

PREVENT USERS FROM CREATING EXTERNAL CONNECTION TO THE INTERNET

In some cases, you might need to prevent users from creating external internet connectivity to their VCN. In the following example policy, the NetworkUsers group is prevented from creating an internet gateway.

```
Allow group NetworkUsers to manage internet-gateways in tenancy
 where request.permission!='INTERNET_GATEWAY_CREATE'
```

PREVENT USERS FROM UPDATING DNS RECORDS AND ZONES

In the following example policy, the NetworkUsers group is prevented from deleting and updating DNS zones and records

```
Allow group NetworkUsers to manage dns-records in tenancy
 where all {request.permission!='DNS_RECORD_DELETE',
            request.permission!='DNS_RECORD_UPDATE'}
Allow group NetworkUsers to manage dns-zones in tenancy
 where all {request.permission!='DNS_ZONE_DELETE',
            request.permission!='DNS_ZONE_UPDATE'}
```

USEFUL CLI COMMANDS

In all the following examples, the environment variables $T , $C and $VCN are set to tenancy OCID, compartment OCID, and VCN OCID, respectively.

LIST OPEN SECURITY LISTS IN A VCN

```
# list open (0.0.0.0/0) security lists in VCN $VCN in compartment $C
oci network security-list list -c $C --vcn-id $VCN | grep "source" | grep "\"0.0.0.0/0\""
```

LIST GATEWAYS IN A VCN

```
# list all internet gateways in VCN $VCN in compartment $C
oci network internet-gateway list -c $C --vcn-id $VCN
# list all DRGs in compartment $C
oci network drg list -c $C
# list all local peering gateways in vcn $VCN in compartment $C
oci network local-peering-gateway list -c $C --vcn-id $VCN
```

LIST ROUTE TABLE RULES IN A VCN

```
# list route table rules in VCN $VCN in compartment $C
oci network route-table list -c $C --vcn-id $VCN
```

Securing Object Storage

SECURITY RECOMMENDATIONS

Assign least privilege access for IAM users and groups to resource types in object-family, namely buckets and objects. For example, the inspect verb gives least privilege and allows checking whether a bucket exists (`HeadBucket`) and list buckets in a compartment (`ListBucket`), while the manage verb gives all permissions. You can craft IAM security policies to give appropriate bucket and object access to various IAM groups. For more information about IAM verbs and permissions for object storage buckets and objects, see Details for Object Storage, Archive Storage, and Data Transfer. For users without IAM credentials, Oracle recommends pre-authenticated requests (PARs) to give time-bound access to objects or buckets.

PUBLIC BUCKETS SECURITY CONTROLS

- A public bucket allows unauthenticated and anonymous reads to all objects in the bucket. Carefully evaluate the intended use case for public buckets before you enable public buckets. Oracle recommends that you use pre-authenticated requests (PAR) to give bucket or object access(read or write) to users without IAM credentials. Note that buckets are created with no public access by default (with access type set to `NoPublicAccess`).
- You can make existing buckets public by updating the bucket access type to `ObjectRead` or `ObjectReadWithoutList`". To minimize possibility of existing buckets being made public inadvertently or maliciously, `BUCKET_UPDATE` permission should be restricted to a minimal set of IAM groups.

PRE-AUTHENTICATED REQUEST (PAR)

- Pre-authenticated requests (PARs) provide a mechanism to provide access to objects stored in buckets, to users who do not have IAM user credentials. In a PAR, an IAM user

who has appropriate privileges for accessing objects, can create URLs which grant time-bound read or write access to these objects. For more information about creating PARs, see Using Pre-Authenticated Requests.

- The creator of a PAR must have `PAR_MANAGE` IAM permission. Following types of PARs can be created, (a) bucket PAR to allow writes to the bucket, (b) object for reading object, (c) object PAR for writing object, and (d) object PAR to read or write an object. PAR cannot be used to list objects in a bucket.
- All PAR accesses to a bucket or object are logged in Audit logs
- Oracle recommends that you note down the PAR URL created. By design, it is not possible to retrieve a forgotten PAR URL. If you forget a PAR URL, you must create a new PAR.

DATA DURABILITY

- To minimize loss of data due to inadvertent deletes by an authorized user or malicious deletes, it is recommended to to give `BUCKET_DELETE` and `OBJECT_DELETE` permissions to a minimum possible set of IAM users/groups. `DELETE` permissions should preferably be given only to tenancy and compartment admins.
- Write once read many (WORM) compliance requires objects can neither be deleted or modified. This can be achieved by granting `OBJECT_CREATE`, `OBJECT_READ`, and `OBJECT_INSPECT` permissions to an IAM group. Note that we precluded `OBJECT_OVERWRITE` permission to prevent modifications to existing object, along with `OBJECT_DELETE`.

DATA ENCRYPTION

- All data in object storage is encrypted at rest by using AES-256. Encryption is on by default and cannot be turned off. Each object is encrypted with its encryption key, and the object encryption keys are encrypted with a master encryption key. In addition, customers can use client-side encryption to encrypt objects with their encryption keys before storing them in object store buckets. An available option for customers is to use the S3 compatibility API, along with client-side object encryption support available in AWS SDK for Java. More details on this SDK are available here (link: https://docs.cloud.oracle.com/iaas/Content/Object/Tasks/s3compatibleapi.htm).

- Data in transit between customer clients (for example, SDKs and CLIs) and object storage public end-points is encrypted with TLS 1.2 by default. FastConnect public peering allows on-premises access to object storage to go over a private circuit, rather than than public internet.

DATA INTEGRITY

- Cryptographic hash using MD5 is provided for all objects uploaded to Object Storage, to verify object data integrity. Oracle recommends that you verify the offline MD5 hash of the object matches the hash value returned by the console or API after the object is uploaded. Oracle Cloud Infrastructure provides the object hash value in base64 encoding. To covert the base64 encoded hash value to hexadecimal, the following command can be used.

  ```
  python -c 'print "BASE64-ENCODED-MD5-VALUE".decode("base64").encode("hex")'
  ```

  Linux provides md5sum command-line utility to compute MD5 hash value of a file in hexadecimal format.
- Object Storage service supports multipart uploads for more efficient and resilient uploads, especially for large objects. In multi-part upload, a large object (to be uploaded to Object Storage) is broken up into smaller parts (by specifying part size in MB), and each part is uploaded separately where Object Storage combines all the parts to put together the original object. If any of the parts fail to upload, only those parts need to be retried for upload, and not the entire object which is very efficient. In a multi-part upload, the MD5 hash values are computed for each part, and a MD5 hash computed over all the individual hash values to get the output MD5 value. In order to verify the MD5 value returned for a multi-part upload, we need to follow the same process for offline MD5 hash calculation. A sample script for offline calculation of MD5 hash value for a multi-part upload to Object Storage is available here (link: https://gist.github.com/itemir/f5bc9fded6483cd79c89ebf4ca1cfd30).

SECURITY POLICY EXAMPLES

In all the examples below, the policies are scoped to a tenancy. However, by specifying a compartment name, they can be scoped down to specific compartment(s) in a tenancy.

RESTRICT GROUP ACCESS TO SPECIFIC BUCKETS

You can restrict access by a group to a specific bucket by using the specific bucket name (target.bucket.name), regular expression matching (/*name/, /name*/, /*name*/), or defined tags (target.tag.definition.name).

The following is an example of restricting access by groups `BucketUsers` to a specific bucket.

```
Allow group BucketUsers to use buckets in tenancy
 where target.bucket.name='BucketFoo'.
```

You can modify this policy to restrict access by group `BucketUsers` to all buckets whose names are prefixed with `ProjectA_`.

```
Allow group BucketUsers to use buckets in tenancy
 where target.bucket.name=/ProjectA_*/
```

You can also match for post-fix (/*_ProjectA/), or sub-string (/*ProjectA*/).

RESTRICT GROUP ACCESS TO READ OR WRITE TO OBJECTS IN A SPECIFIC BUCKET

The following example allows listing and reading objects by group `BucketUsers` from a specific bucket named `BucketFoo`.

```
Allow group BucketUsers to read buckets in tenancy
Allow group BucketUsers to manage objects in tenancy
 where all {target.bucket.name='BucketFoo',
            any {request.permission='OBJECT_INSPECT',
                 request.permission='OBJECT_READ'}}
```

The following policy modifies the previous policy to allow listing and writing objects to `BucketFoo`.

```
Allow group BucketUsers to read buckets in tenancy
Allow group BucketUsers to manage objects in tenancy
 where all {target.bucket.name='BucketFoo',
            any {request.permission='OBJECT_INSPECT',
                 request.permission='OBJECT_CREATE'}}
```

You can restrct this policy to read or write access to a set of buckets by using regular expressions or tags rather than a specific bucket.

PREVENT DELETE OF BUCKETS OR OBJECTS

In the following example, the group `BucketUsers` can perform all actions on buckets and objects except delete.

```
Allow group BucketUsers to manage objects in tenancy
 where request.permission!='OBJECT_DELETE'
Allow group BucketUsers to manage buckets in tenancy
 where request.permission!='BUCKET_DELETE'
```

The following example further restricts deletes from a specific bucket (`BucketFoo`).

```
Allow group BucketUsers to manage objects in tenancy
  where any {target.bucket.name!='BucketFoo',
             all {target.bucket.name='BucketFoo',
                  request.permission!='OBJECT_DELETE'}}
```

ENABLE WRITE ONCE, READ MANY COMPLIANCE FOR OBJECTS

The following policy enables write once, read many (WORM) compliance by removing permissions for group `BucketUsers` to delete or update objects.

```
Allow group BucketUsers to manage objects in tenancy
 where any {request.permission='OBJECT_INSPECT',
            request.permission='OBJECT_READ',
            request.permission='OBJECT_CREATE'}
```

The following policy allows for WORM compliance.

```
Allow group BucketUsers to manage buckets in tenancy
 where any {request.permission='BUCKET_INSPECT',
            request.permission='BUCKET_READ',
            request.permission='BUCKET_CREATE',
            request.permission='PAR_MANAGE'}
```

PREVENT PUBLIC BUCKETS CONFIGURATION

As mentioned in previous section, `BUCKET_CREATE` and `BUCKET_UDPATE` permissions are required to create public buckets, or update existing buckets to public. Removing these permissions prevents users from creating new or making existing buckets public.

```
Allow group BucketUsers to manage buckets in tenancy
 where any {request.permission='BUCKET_INSPECT',
```

```
            request.permission='BUCKET_READ',
            request.permission='PAR_MANAGE'}
```

USEFUL CLI COMMANDS

In the examples below, environment variables $NS and $C are set tenancy namespace and compartment OCID, respectively.

LIST PUBLIC BUCKETS

The following command returns the public access buckets for each bucket in a namespace.

```
# "public-access-type" of 'NoPublicAccess' indicates a private bucket, and
# anything else ('ObjectRead') indicates a public bucket
oci os bucket get -ns $NS --bucket-name $BUCKET_NAME | grep "public-access-type"
```

LIST BUCKET PRE-AUTHENTICATED REQUESTS (PARS)

```
# list all PARs for objects in bucket $BUCKET_NAME
oci os preauth-request list -ns $NS -bn $BUCKET_NAME
```

Securing Resource Manager

Resource Manager allows you to automate installing and provisioning Oracle Cloud Infrastructure resources by committing the provisioning instructions to configuration files. These configuration files capture the step-by-step provisioning instructions using a declarative language that follows the "infrastructure-as-code" model. The provisioning instructions are executed as "jobs"; the Oracle Cloud Infrastructure resources that are provisioned when you run the jobs are organized into "stacks."

Executing jobs and provisioning stacks is gated using role-based access control (RBAC), which is enabled by Oracle Cloud Infrastructure Identity and Access Management (IAM). This gives administrators granular control over user access to Oracle Cloud Infrastructure resources and the actions that users can take on these resources.

The Resource Manager security scheme rests on three pillars:

- **Security groups**. Administrator-defined groups that have permission to perform specific operations on stacks and jobs. Individual users are assigned to security groups

and can then perform operations that are allowed by that group.

- **Permission sets**. Sets of permissions that are specific to jobs and stacks. Permission sets for jobs and stacks are listed in Table 1.
- **Operations**. The operations (or actions) that are allowed and the permissions that are required to perform each one. These are listed in Table 2.

Resource Manager Operations and Permissions

The Resource Manager supports two permission sets: one for stack resources and another for job resources. Table 1 lists permission sets that are associated with each resource type.

Table 1. Resource Types and Permission Sets

Resource Type	Permissions
Stacks (`orm-stacks`)	inspect orm-stackread orm-stack use orm-stack create orm-stack update orm-stack delete orm-stack
Jobs (`orm-jobs`)	inspect orm-job read orm-job manage orm-job

Each of the permissions in the preceding table are associated with specific Resource Manager operations. The table that follows lists Resource Manager operations, then shows which permissions are required to execute each of the operations. Notice that the `CreateJob` operation requires two permissions.

TABLE 2. RESOURCE OPERATIONS AND REQUIRED PERMISSIONS

Operation	Permission
Generate a list of stacks (`ListStacks`)	inspect orm-stack
Create a stack (`CreateStack`)	create orm-stack
Get a stack (`GetStack`)	read orm-stack
Update a stack (`UpdateStack`)	update orm-stack
Delete a stack (`DeleteStack`)	delete orm-stack
Get a stack Terraform configuration (`GetStackTfConfig`)	read orm-stack
List jobs (`ListJobs`)	inspect orm-job
Create a job (`CreateJob`)	use orm-stack *and* manage orm-job
Get a job (`GetJob`)	read orm-jobs
Update a job (`UpdateJob`)	manage orm-job
Cancel a job (`CancelJob`)	manage orm-job
Get a job Terraform state file (`GetJobTfState`)	read orm-job
Get a job Terraform configuration (`GetJobTfConfig`)	read orm-job
Get a job Terraform execution plan (`GetJobTfExecutionPlan`)	read orm-job
Get job logs (`GetJobLogs`)	read orm-job

RECOMMENDED SECURITY POLICIES

Following are policy recommendations for securing Resource Manager. For more information about security policies, see Getting Started with Policies. See also How Policies Work and Policy Syntax.

Permission to Manage Stacks and Jobs

The following policy grants permission to a specified group to manage both stacks and jobs, and also to manage resources on the tenancy stacks.

```
Allow group <group_name> to manage orm-stacks in compartment
Allow group <group_name> to manage orm-jobs in compartment
```

Prevent Users from Running Destroy Jobs

In addition to granting permission to perform specific operations, you can also create a policy that prevents certain actions. The following policy example explicitly prevents members of a specified group from running Destroy jobs on a stack.

```
Allow group <group_name> to use orm-stacks in compartment
Allow group <group_name> to read orm-jobs in compartment
Allow group <group_name> to manage orm-jobs in compartment where any {target.job.operation = 'PLAN',
target.job.operation = 'APPLY'}
```

Notice that you must include the new permission to `read orm-jobs in compartment` because the statement includes a `where` condition that references variables that are not relevant to listing or getting jobs.

Potential Security Risks and Mitigations

Terraform State Files

Terraform state (.tfstate) can contain sensitive data, including resource IDs and in some cases sensitive user data like passwords. HashiCorp provides recommendations for handling Terraform state in the article Sensitive Data in State.

To control access to the Terraform state file, you can create a security policy that limits access to reading jobs, such as the following:

```
Allow group <group_name> to read orm-jobs in compartment
```



Note

Because the permission **read orm-jobs** also affects other operations such as getting logs and Terraform configurations, you should segregate state files in a compartment on which a restrictive policy will not limit the ability to perform other operations.

TERRAFORM CONFIGURATION .ZIP FILES

The Resource Manager workflow includes uploading your Terraform configuration to the service as a .zip file. Because the Terraform configuration can be accessed using the ORM API (`GetJobTfConfig`), it is highly recommended that you do not include sensitive information in your configuration files.

Addressing Basic Configuration Issues

This topic lists procedures to address common configuration issues that affect the security of your cloud resources.

Block Volume

Block volume detached from instance

Issue: Ensure that only Oracle Cloud Infrastructure administrators can detach block volumes from instances.

Basics: When you detach a block volume it decouples the volume from its associated instance, affecting the data available to the instance. This could impact data availability from business-critical data to the successful completion of scheduled volume backups. To minimize loss of data due to inadvertent volume detachments by an authorized user or malicious

volume detachments you should restrict the `VOLUME_ATTACHMENT_DELETE` permission to administrators.

To prevent detachment of block volumes:

The following policy allows the group `VolumeUsers` to manage volumes and volume attachments except for detaching volumes:

```
Allow group VolumeUsers to manage volumes in tenancy
Allow group VolumeUsers to manage volume-attachments in tenancy
    where request.permission!='VOLUME_ATTACHMENT_DELETE'
```

This change prevents `VolumeUsers` from detaching volumes from instances.

More information:

- Securing Block Volume
- Getting Started with Policies
- How Policies Work
- For volume-family Resource Types

Compute

Instance created based on unapproved custom image

Issue: An instance was created from a custom image that is unsupported in your environment.

Basics: When users create instances they can select from Oracle-provided images, boot volumes from terminated instances, or custom images. Custom images represent a wide variety of images which can include images that aren't approved for your environment. If you use tags in your Oracle Cloud Infrastructure tenancy to identify approved images, verify whether the image the instance is based on is an approved image and terminate the instance if necessary.

To verify the tags for the image the instance was created from:

The following procedure is for the Oracle Cloud Infrastructure Console.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
2. Click the instance you're interested in.
3. Click the **Image** link to view the source image.
4. Click the **Tags** tab to view the tags applied to this image.

If the custom image does not have an approved tag, and the instance needs to be terminated, see Terminating an Instance

More information:

- Securing Compute
- Resource Tags
- Creating an Instance
- Image Import/Export
- Bring Your Own Image (BYOI)

IAM

Member of the Administrators group used API keys

Issue: A user who is a member of the Administrators group accessed resources using an API key.

Basics:

- API keys are credentials used to grant programmatic access to Oracle Cloud Infrastructure.
- For security and governance reasons, users should only have access to resources they need to interact with.

- For individuals who are members of the Administrators group who also need access to resources through the API, create another user in IAM to which you attach the API keys. Grant the user with the API keys permissions to only the resources they need to interact with programmatically.

To create a user, group, and policy with limited permissions:

The following set of procedures shows you how to set up an example user with limited permissions. In this example, the user needs to be able to launch instances in a specific compartment.

The following procedure is for the Oracle Cloud Infrastructure Console.

Create a User

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Users**.
2. Click **Create User**.
3. In the **New User** dialog:
    - **Name**: Enter a unique name or email address for the new user.
      The value will be the user's login to the Console and must be unique across all other users in your tenancy.
    - **Description**: Enter a description (required).
4. Click **Create User**.

Create a Group

Next, create the group ("InstanceLaunchers") that you will create the policy for.

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Groups**.
2. Click **Create Group**.

3. In the **Create Group** dialog:
    - **Name:** Enter a unique name for your group, for example, "InstanceLaunchers". Note that the name cannot contain spaces.
    - **Description:** Enter a description (required).
4. Click **Create Group**.

Create a Policy

In this example, the policy grants members of the group InstanceLaunchers permissions to launch instances in a specific compartment (CompartmentA).

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Policies**.
2. Click **Create Policy**.
3. Enter a unique **Name** for your policy, for example, "InstanceLaunchersPolicy". Note that the name cannot contain spaces.
4. Enter a **Description** (required), for example, "Grants users permission to launch instances in CompartmentA".
5. Enter the following **Statement**:

    ```
    Allow group InstanceLaunchers to manage instance-family in compartment CompartmentA
    ```

    This statement grants members of the InstanceLaunchers group permissions to launch and manage instances in the compartment called CompartmentA.
6. Click **Create Policy**.

Add the User to the Group

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity**

and click **Users**.

2. In the **Users** list, find the user and click the name.
3. On the user detail page, click **Groups** (on the left side of the page). The list of groups that the user belongs to is displayed.
4. Click **Add User to Group.**
5. From the **Groups** list, select InstanceLaunchers.
6. Click **Add**.

Upload an API signing key for the user

The following procedure works for a regular user or an administrator. Administrators can upload an API key for either another user or themselves.



Important

The API key must be an **RSA key in PEM format (minimum 2048 bits)**. The PEM format looks something like this:

```
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAoTFqF...
...
-----END PUBLIC KEY--
```

For more information about generating a public PEM key, see Required Keys and OCIDs.

1. View the user's details:
   - If you're uploading an API key for *yourself*: Open the **User** menu () and click **User Settings**.

- If you're an administrator uploading an API key for *another user*: In the Console, click **Identity**, and then click **Users**. Locate the user in the list, and then click the user's name to view the details.

2. Click **Add Public Key**.
3. Paste the key's value into the window and click **Add**.
   The key is added and its fingerprint is displayed (example fingerprint: d1:b2:32:53:d3:5f:cf:68:2d:6f:8b:5f:77:8f:07:13).

More information:

- Securing IAM
- How Policies Work and Common Policies
- Managing User Credentials
- Managing Groups
- Managing Users

Policy grants broad permissions

Issue: A policy grants full management permissions for at least one service in a compartment or in the tenancy.

Basics:

- Access to resources is controlled through policies. A *policy* is a document that specifies who can access which Oracle Cloud Infrastructure resources that your company has, and how. A policy simply allows a group to work in certain ways with specific types of resources in a particular compartment.
- For security and governance reasons, users should only have access to resources they need.
- Consider carefully the access level a user needs. Policy language provides a default set of verbs (`manage`, `use`, `read`, `inspect`) that allow you to easily scope users' permissions

to a set of common tasks. For example, if a user needs to be able to update resources, but does not need to create or delete them, grant them the `use` permission, rather than the `manage` permission.

- The policy language is designed to let you write simple statements involving only verbs and resource-types, without having to state the permissions in the statement. For more fine-grained access control, you can use conditions combined with permissions or API operations to reduce the scope of access granted by a particular verb.
- Wherever possible, scope access to the specific compartments a user needs access to, rather than scoping it to the full tenancy.

Tips for writing least-privilege policies:

Scope the policy to a compartment instead of the tenancy

Each policy consists of one or more policy *statements* that follow a basic syntax. Where possible, scope policies to compartments, rather than to the tenancy. For example, update a policy like this:

```
Allow group <group_name> to <verb><resource-type> in tenancy
```

to include just the compartments needed:

```
Allow group <group_name> to <verb><resource-type> in compartment <compartment_name>
```

If the user needs access to multiple compartments, create a policy statement for each compartment. It is then easy to remove access to individual compartments, if necessary.

Scope permissions to those required to perform a job function

Oracle defines the possible verbs you can use in your policies. Here's a summary of the verbs, from least amount of access to the most:

Verb	Types of Access Covered	Target User
`inspect`	Ability to list resources, without access to any confidential information or user-specified metadata that may be part of that resource.	Third-party auditors
`read`	Includes `inspect` plus the ability to get user-specified metadata and the actual resource itself.	Internal auditors
`use`	Includes `read` plus the ability to work with existing resources (the actions vary by resource type). In general, this verb does not include the ability to create or delete that type of resource.	Day-to-day end users of resources
`manage`	Includes all permissions for the resource.	Administrators

Users who don't need to create or delete resources generally don't need manage permissions. If you have a policy like

```
Allow group <group_name> to manage <resource-type> in compartment <compartment_name>
```

but the user will never create or delete the resource-type, consider rewriting the policy to

```
Allow group <group_name> to use <resource-type> in compartment <compartment_name>
```

The Policy Reference includes details of the specific resource-types for each service, and which verb + resource-type combination gives access to which API operations.

Service-specific links

- Details for the Audit Service
- Details for Container Engine for Kubernetes
- Details for the Core Services (this includes Networking, Compute, and Block Volume)
- Details for the Database Service

- Details for the DNS Service
- Details for the Email Service
- Details for the File Storage Service
- Details for IAM
- Details for Load Balancing
- Details for Object Storage, Archive Storage, and Data Transfer
- Details for Registry
- Details for the Search Service

For fine-grained access control, scope access using conditions and API operations

In a policy statement, you can use conditions combined with permissions or API operations to reduce the scope of access granted by a particular verb.

For example, let's say you want group XYZ to be able to list, get, create, or update groups (change their description), but not delete them. To list, get, create, and update groups, you need a policy with `manage groups` as the verb and resource-type. According to the table in Details for Verbs + Resource-Type Combinations, the permissions covered are:

- GROUP_INSPECT
- GROUP_UPDATE
- GROUP_CREATE
- GROUP_DELETE

To restrict access to only the desired permissions, you could add a condition *that explicitly states the permissions you want to allow*:

```
Allow group XYZ to manage groups in tenancy

 where any {request.permission='GROUP_INSPECT',
```

```
            request.permission='GROUP_CREATE',
            request.permission='GROUP_UPDATE'}
```

An alternative would be a policy that *allows all permissions except* GROUP_DELETE:

```
Allow group XYZ to manage groups in tenancy where request.permission != 'GROUP_DELETE'
```

Another alternative would be to write a condition *based on the specific API operations*. Notice that according to the table in Permissions Required for Each API Operation, both `ListGroups` and `GetGroup` require only the GROUP_INSPECT permission. Here's the policy:

```
Allow group XYZ to manage groups in tenancy

 where any {request.operation='ListGroups',
            request.operation='GetGroup',
            request.operation='CreateGroup',
            request.operation='UpdateGroup'}
```

It can be beneficial to use permissions instead of API operations in conditions. In the future, if a new API operation is added that requires one of the permissions listed in the permissions-based policy above, that policy will already control XYZ group's access to that new API operation.

But notice that you can further scope a user's access to a permission by *also* specifying a condition based on API operation. For example, you could give a user access to GROUP_INSPECT, but then only to `ListGroups`.

```
Allow group XYZ to manage groups in tenancy

 where all {request.permission='GROUP_INSPECT',
            request.operation='ListGroups'}
```

More information:

- Securing IAM
- How Policies Work and Common Policies
- Advanced Policy Features
- Managing Policies

API signing keys over 90 days old

Issue: A user's API signing keys are older than 90 days. Oracle recommends that you rotate API keys at least every 90 days.

Basics:

- API keys are credentials used to grant programmatic access to Oracle Cloud Infrastructure.
- It is a security engineering best practice and compliance requirement to rotate API keys regularly, every 90 days or less.
- Ensure that you test the new keys before you deactivate the old keys.

To generate and upload new API keys:

The following procedure is for the Oracle Cloud Infrastructure Console.

Generate new API keys

You can use the following OpenSSL commands to generate the key pair in the required PEM format. If you're using Windows, you'll need to install Git Bash for Windows and run the commands with that tool.

1. If you haven't already, create a `.oci` directory to store the credentials:

   ```
   mkdir ~/.oci
   ```

2. Generate the private key with one of the following commands.

   - Recommended: To generate the key, encrypted with a passphrase you provide when prompted:

     ```
     openssl genrsa -out ~/.oci/oci_api_key.pem -aes128 2048
     ```

     Note: For Windows, you may need to insert `-passout stdin` to be prompted for a passphrase. The prompt will just be the blinking cursor, with no text.

     ```
     openssl genrsa -out ~/.oci/oci_api_key.pem -aes128 -passout stdin 2048
     ```

- To generate the key with no passphrase:

```
openssl genrsa -out ~/.oci/oci_api_key.pem 2048
```

3. Ensure that only you can read the private key file:

```
chmod go-rwx ~/.oci/oci_api_key.pem
```

4. Generate the public key:

```
openssl rsa -pubout -in ~/.oci/oci_api_key.pem -out ~/.oci/oci_api_key_public.pem
```

Note: For Windows, if you generated the private key with a passphrase, you may need to insert `-passin stdin` to be prompted for the passphrase. The prompt will just be the blinking cursor, with no text.

```
openssl rsa -pubout -in ~/.oci/oci_api_key.pem -out ~/.oci/oci_api_key_public.pem -passin stdin
```

5. Copy the contents of the public key to the clipboard using pbcopy, xclip or a similar tool (you'll need to paste the value into the Console later). For example:

```
cat ~/.oci/oci_api_key_public.pem | pbcopy
```

Your API requests will be signed with your private key, and Oracle will use the public key to verify the authenticity of the request. You must upload the public key to IAM (instructions below).

Get the key's fingerprint

You can get the key's fingerprint with the following OpenSSL command. If you're using Windows, you'll need to install Git Bash for Windows and run the command with that tool.

```
openssl rsa -pubout -outform DER -in ~/.oci/oci_api_key.pem | openssl md5 -c
```

When you upload the public key in the Console, the fingerprint is also automatically displayed there. It looks something like this: 12:34:56:78:90:ab:cd:ef:12:34:56:78:90:ab:cd:ef

Upload the API signing key for the user

You can upload the PEM public key in the Console, located at https://console.us-ashburn-

1.oraclecloud.com. If you don't have a login and password for the Console, contact an administrator.

1. Open the Console, and sign in.
2. View the details for the user who will be calling the API with the key pair:
    - If you're signed in as this user, click your username in the top-right corner of the Console, and then click **User Settings**.
    - If you're an administrator doing this for another user, instead click **Identity**, click **Users**, and then select the user from the list.
3. Click **Add Public Key**.
4. Paste the contents of the PEM public key in the dialog box and click **Add**.

The key's fingerprint is displayed (for example, 12:34:56:78:90:ab:cd:ef:12:34:56:78:90:ab:cd:ef).

Notice that after you've uploaded your first public key, you can also use the UploadApiKey API operation to upload additional keys. You can have up to three API key pairs per user. In an API request, you specify the key's fingerprint to indicate which key you're using to sign the request.

Test the new key

Test the key in a sample API call against Oracle Cloud Infrastructure.

Delete the old key

The following procedure works for a regular user or an administrator. Administrators can delete an API key for either another user or themselves.

1. View the user's details:
    - If you're deleting an API key for *yourself*: Open the **User** menu () and click **User Settings**.

- If you're an administrator deleting an API key for *another user*: In the Console, click **Identity**, and then click **Users**. Locate the user in the list, and then click the user's name to view the details.

2. For the API key you want to delete, click **Delete**.
3. Confirm when prompted.

The API key is no longer valid for sending API requests.

More information:

- Securing IAM
- API Signing Key

Tenancy administrator privilege grant to an IAM group

Issue: A group other than the Administrators group has been granted administrator privileges.

Basics:

- Granting the tenancy administrator privilege (`manage all-resources in tenancy`) to a group enables the members to have full access to all resources in the tenancy.
- This high-privilege entitlement must be controlled and restricted to only the users who need it to perform their job function.
- Verify with the Oracle Cloud Infrastructure administrator that this entitlement grant was sanctioned and that the membership of the group remains valid after the grant of the administrator privilege.
- Rather than create an alternative group with administrator privileges, consider instead adding users needing administrator privileges to the default Administrators group.

To resolve this issue:

Add users who need administrator privileges to the Administrators group:

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Groups**.
2. In the **Groups** list, click **Administrators**.
3. Click **Add User to Group**.
4. In the **Add User to Group** dialog, select the user from the **User** list.
5. Click **Add User**.

Remove the policy or policy statement that grants the (non-Administrators) group administration privileges.

1. Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Policies**.
   A list of the policies in the compartment you're viewing is displayed. If you don't see the one you're looking for, verify that you're viewing the correct compartment (select from the list on the left side of the page).
2. Click the policy you want to update.
   The policy's details and statements are displayed.
3. Find the statement that grants administrator privileges to the group. This policy will look like:

   ```
   Allow group <group_name> to manage all-resources in tenancy
   ```

   Click the the Actions icon (three dots) and then click **Delete**.
4. If the policy has no other statements, you can delete the policy by clicking **Delete** next to the policy name.

More information:

- Securing IAM
- Managing Policies

Networking: VCN, Load Balancers, and DNS

No ingress rules in security lists

Issue: A VCN's security lists have no ingress rules. This means that the instances in the VCN can't receive incoming traffic.

Basics:

- Security lists provide stateful and stateless firewall capability to control network access to your instances.
- A security list is configured at the subnet level and enforced at the instance level.
- You can associate multiple security lists with a subnet. A packet is allowed if it matches any rule in any of the security lists used by the subnet.
- If there are no ingress (inbound) rules in any of the subnet's security lists, no traffic is allowed to the instances in that subnet.
- For defense in depth, ingress security list rules should state a specific known source and not an open source (0.0.0.0/0).
- You can configure an exception in Oracle CASB Cloud Service to reduce alerts from exempted security lists.

To add an ingress rule to an existing security list:

The following procedure is for the Oracle Cloud Infrastructure Console.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Confirm you're viewing the compartment that contains the cloud network you're interested in.
3. Click the cloud network you're interested in.
4. Click **Security Lists**.
5. Click the security list you're interested in.

6. Click **Edit All Rules**.
7. Add at least one ingress rule:
    a. In the **Allow Rules for Ingress** section, click **+ Rule**.
    b. Choose whether it's a stateful or stateless rule (see Stateful vs. Stateless Rules). By default, rules are stateful unless you specify otherwise.
    c. Enter the source CIDR. Typical CIDRs you might specify in a rule are the CIDR block for your on-premises network or a particular subnet. If you're setting up a security list rule to allow traffic with a service gateway, instead see Task 3: (Optional) Update the security lists for the subnet.
    d. Select the protocol (for example, TCP, UDP, ICMP, "All protocols", and so on).
    e. Enter further details depending on the protocol:
        - If you chose TCP or UDP, enter a source port range and destination port range. You can enter "All" to cover all ports. If you want to allow a specific port, enter the port number (for example, 22 for SSH or 3389 for RDP) or a port range (for example, 20-22).
        - If you chose ICMP, you can enter "All" to cover all types and codes. If you want to allow a specific ICMP type, enter the type and an optional code separated by a comma (for example, 3,4). If the type has multiple codes you want to allow, create a separate rule for each code.
8. When you're done, click **Save Security List Rules**.

This change enables ingress access from the source CIDR block listed in the rule. Add additional rules if you want to allow ingress from other known sources.

More information:

- Securing Networking: VCN, Load Balancers, and DNS
- Security Lists
- UpdateSecurityList

Security list allows traffic from any IP address (open source)

Issue: A security list has at least one rule with an open source (0.0.0.0/0). This means that traffic could come from any source and is not controlled.

Basics:

- Security lists provide stateful and stateless firewall capability to control network access to your instances.
- A security list is configured at the subnet level and enforced at the instance level.
- You can associate multiple security lists with a subnet. A packet is allowed if it matches any rule in any of the security lists used by the subnet.
- If there are no ingress (inbound) rules in any of the subnet's security lists, no traffic is allowed to the instances in that subnet.
- For defense in depth, ingress security list rules should state a specific known source and not an open source (0.0.0.0/0).
- You can configure an exception in Oracle CASB Cloud Service to reduce alerts from exempted security lists.

To change the source of a security list rule:

The following procedure is for the Oracle Cloud Infrastructure Console.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Confirm you're viewing the compartment that contains the cloud network you're interested in.
3. Click the cloud network you're interested in.
4. Click **Security Lists**.
5. Click the security list you're interested in.
6. Click **Edit All Rules**.

7. Locate the rule that lists 0.0.0.0/0 as the source CIDR.
8. For that rule, change 0.0.0.0/0 to the CIDR block of a known source.
9. Click **Save Security List Rules**.

This change restricts ingress so packets are allowed only from a specific CIDR block. Add additional rules if you want to allow ingress from other known sources.

More information:

- Securing Networking: VCN, Load Balancers, and DNS
- Security Lists
- UpdateSecurityList

Security list allows traffic to sensitive ports

Issue: A security list has at least one rule that enables access to a sensitive port.

Basics:

- Security lists provide stateful and stateless firewall capability to control network access to your instances.
- A security list is configured at the subnet level and enforced at the instance level.
- You can associate multiple security lists with a subnet. A packet is allowed if it matches any rule in any of the security lists used by the subnet.
- If there are no ingress (inbound) rules in any of the subnet's security lists, no traffic is allowed to the instances in that subnet.
- For defense in depth, ingress security list rules should state a specific known source and not an open source (0.0.0.0/0).
- You can configure an exception in Oracle CASB Cloud Service to reduce alerts from exempted security lists.

Recommendation: Update the subnet's security list to enable access to instances through SSH (TCP port 22) or RDP (TCP port 3389) on a temporary, as-needed basis, and only from authorized CIDR blocks (not 0.0.0.0/0). To perform instance health checks, update the security list to allow ICMP pings.

To change an existing security list:

The following procedure is for the Oracle Cloud Infrastructure Console.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Confirm you're viewing the compartment that contains the cloud network you're interested in.
3. Click the cloud network you're interested in.
4. Click **Security Lists**.
5. Click the security list you're interested in.
6. Click **Edit All Rules**.
7. Make one or more of these changes:
    - Delete an existing rule by clicking the **X** next to the rule.
    - Change an existing rule in the list. For example: change the source from 0.0.0.0/0 to the CIDR block of a known source.
    - Add a new rule by clicking **+ Rule** and entering values for the new rule.
8. Click **Save Security List Rules**.

More information:

- Securing Networking: VCN, Load Balancers, and DNS
- To enable RDP access
- Security Lists
- UpdateSecurityList

Internet gateway attached to VCN

Issue: A VCN has an internet gateway. The gateway must be authorized to be attached to the VCN and must not unintentionally expose resources to the internet.

Basics:

- Gateways provide external connectivity to hosts in a VCN. For example: an internet gateway enables direct internet connectivity for instances that are in a public subnet and have a public IP address. A dynamic routing gateway (DRG) enables connectivity with the on-premises network over an IPSec VPN or FastConnect.
- To enable traffic through the internet gateway from a particular subnet in the VCN, there must be a rule in the subnet's route table that lists the internet gateway as a route target. To delete the internet gateway from the VCN, you must first delete any route rules that specify the internet gateway as the target.
- You can configure an exception in Oracle CASB Cloud Service to reduce alerts from exempted VCNs.

To remove an internet gateway from a VCN:

Prerequisite: Ensure that there are no route rules that specify the internet gateway as a target.

The following procedure is for the Oracle Cloud Infrastructure Console.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Confirm you're viewing the compartment that contains the cloud network you're interested in.
3. Click the cloud network you're interested in.
4. Click **Internet Gateways**.
5. Click the Actions icon (three dots) for the internet gateway, and then click **Terminate**.
6. Confirm when prompted.

This change disables direct internet connectivity for the VCN.

More information:

- Securing Networking: VCN, Load Balancers, and DNS
- Internet Gateway
- DeleteInternetGateway
- Public IP Addresses

Instance has a public IP

Issue: An instance has a public IP address. This means the instance could be publicly addressable if other required components are present and configured correctly in the VCN.

Basics:

- Carefully consider allowing internet access to any instances. For example, don't accidentally allow internet access to sensitive DB systems.
- For an instance to be publicly addressable:
    - The instance must have a public IP address and reside in a public subnet in the VCN (instances in private subnets cannot have public IP addresses).
    - The subnet's security list must be configured to allow traffic for all IPs (0.0.0.0/0) and all ports.
    - The VCN must have an internet gateway and be configured to route outbound traffic from the subnet to the internet gateway.
- An instance can have more than one public IP address. A given public IP is assigned to a private IP on a particular VNIC on the instance. An instance can have more than one VNIC, and each VNIC can have more than one private IP.

To remove a public IP address from an instance:

The following procedure is for the Oracle Cloud Infrastructure Console.

1. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Instances**.
2. Confirm you're viewing the compartment that contains the instance you're interested in.
3. Click the instance to view its details.
4. Click **Attached VNICs**.
   The primary VNIC and any secondary VNICs attached to the instance are displayed.
5. Click the VNIC you're interested in.
   The VNIC's primary private IP and any secondary private IPs are displayed.
6. For the private IP you're interested in, click the Actions icon (three dots), and then click **Edit**.
7. In the **Public IP Address** section, for **Public IP Type**, select the radio button for **No Public IP**.
8. Click **Update**.

The public IP is unassigned from the instance.

More information:

- Securing Networking: VCN, Load Balancers, and DNS
- Public IP Addresses
- DeletePublicIp
- Internet Gateway

Load balancer has no inbound rules or listeners

Issue: A load balancer's subnet security lists have no ingress rules, or a load balancer has no listener. In this case, the load balancer can't receive incoming traffic.

Basics:

- Load balancers provide automated traffic distribution from one entry point to multiple servers reachable from your virtual cloud network (VCN). Each load balancer exists in a

subnet governed by security list rules. A load balancer receives incoming data traffic from one or more listeners.

- Security lists provide stateful and stateless firewall capability to control network access to your load balancer and backend servers.
    - If there are no ingress (inbound) rules in any of the subnet's security lists, no traffic is allowed to the instances in that subnet.
    - For defense in depth, configure ingress security list rules to state a specific known source and not an open source (0.0.0.0/0).
- A listener is a logical entity that checks for incoming traffic on the load balancer's IP address.
    - To handle TCP, HTTP, and HTTPS traffic, you must configure at least one listener per traffic type.
    - You can apply path route rules to a listener to route traffic to the correct backend set without using multiple listeners or load balancers. A *path route* is a string that the listener matches against an incoming URI to determine the appropriate destination backend set.
- Ensure that your Oracle Cloud Infrastructure load balancers use inbound rules or listeners to allow access only from known resources.
- Exceptions can be configured in CASB to reduce alerts from exempted load balancers.

To enable a listener to accept traffic:

The following procedure is for the Oracle Cloud Infrastructure Console.

To enable a listener to accept traffic, you must update your VCN's security list rules:

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
   The list of VCNs in the current compartment appears.
2. Click the name of the VCN containing your load balancer, and then click **Security Lists**.
   A list of the security lists in the cloud network appears.

3. Click the name of the security list that applies to your load balancer.
4. Click **Edit All Rules**.
5. Under **Allow Rules for Ingress**, click **Add Rule**.
6. Configure the ingress rule to allow access from known resources only.
   For details on ingress rule configuration, see Parts of a Security List Rule.
7. Click **Save Security List Rules**.

To create a listener:

Usually, you create a listener as part of the load balancer creation workflow. To create a listener for an existing load balancer:

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
2. Choose the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
3. In the **Resources** menu, click **Listeners** (if necessary), and then click **Create Listener**.
4. In the **Create Listener** dialog box, enter the following:
   - **Name:** Required. Specify a friendly name for the listener. The name must be unique, and cannot be changed. Avoid entering confidential information.
   - **Hostname:** Optional. Select up to 16 virtual hostnames for this listener.



Important

To apply a virtual hostname to a listener, the name must be part of the load balancer's configuration. If the load balancer has no associated hostnames, you can create one on the **Hostnames** page.

- **Protocol:** Required. Specify the protocol to use, either HTTP or TCP.
- **Port:** Required. Specify the port on which to listen for incoming traffic.
- **Use SSL:** Optional. Check this box to associate an SSL certificate bundle with the listener. The following settings are required to enable SSL handling. See Managing SSL Certificates for more information.
    - **Certificate Name:** Required. The friendly name of the SSL certificate bundle to use.
    - **Verify Peer Certificate:** Optional. Select this option to enable peer certificate verification.
    - **Verify Depth:** Optional. Specify the maximum depth for certificate chain verification.
- **Backend Set:** Required. Specify the default backend set to which the listener routes traffic.
- **Timeout in seconds:** Optional. Specify the maximum idle time in seconds. This setting applies to the time allowed between two successive receive or two successive send network input/output operations during the HTTP request-response phase.



Tip

The maximum value is 7200 seconds. For more information, see Connection Management.

- **Path Route Set:** Optional. Specify the name of the set of path-based routing rules that applies to this listener's traffic.



Important

- To apply a path route set to a listener, the set must be part of the load balancer's configuration.
- To remove a path route set from an existing listener, choose **None** as the **Path Route Set** option. The path route set remains available for use by other listeners on this load balancer.

- **Rule Sets:** Optional. Select a rule set to apply to this listener's traffic.



Important

- To apply a rule set to a listener, the set must be part of the load balancer's configuration.
- To remove a rule set from the list, click the corresponding red box. The rule set remains available for use by other listeners on this load balancer.

5. Click **Create**.

When you create a listener, you must also update your VCN's security list rules to allow traffic to that listener.

More information:

- Securing Networking: VCN, Load Balancers, and DNS
- Security Lists

- Managing a Load Balancer
- Managing Load Balancer Listeners
- Managing Request Routing

Load balancer has no backend sets

Issue: A load balancer has no backend set. In this case, the load balancer has no place to distribute incoming data and no means to monitor backend server health.

Basics:

- A backend set is a logical entity defined by a load balancing policy, a health check policy, and a list of backend servers.
- The backend set determines the load balancer's traffic distribution policy, such as:
    - IP Hash
    - Least Connections
    - Weighted Round Robin
- You specify the test parameters to confirm the health of backend servers when you create a backend set.
- If you have an existing load balancer with no backend set, you can specify the backend servers that receive traffic from the load balancer after you create a backend set.
- You can configure an exception in Oracle CASB Cloud Service to reduce alerts from exempted load balancers.

To create a backend set:

The following procedure is for the Oracle Cloud Infrastructure Console.

Usually, you create a backend set as part of the load balancer creation workflow. To create a backend set for an existing load balancer:

1. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
2. Click the name of the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
3. In the **Resources** menu, click **Backend Sets** (if necessary), and then click **Create Backend Set**.
4. In the **Create Backend Set** dialog box, enter the following:
    - **Name:** Required. Specify a friendly name for the backend set. It must be unique within the load balancer, and it cannot be changed.

      Valid backend set names include only alphanumeric characters, dashes, and underscores. Backend set names cannot contain spaces. Avoid entering confidential information.
    - **Policy:** Required. Choose the load balancer policy for the backend set. The available options are:
        - **IP Hash**
        - **Least Connections**
        - **Weighted Round Robin**

      For more information on these policies, see How Load Balancing Policies Work.
    - **Use SSL:** Optional. Check this box to associate an SSL certificate bundle with the backend set. The following settings are required to enable SSL handling. See Managing SSL Certificates for more information.
        - **Certificate Name:** Required. The friendly name of the SSL certificate to use. See Managing SSL Certificates for more information.
        - **Verify Peer Certificate:** Optional. Select this option to enable peer certificate verification.
        - **Verify Depth:** Optional. Specify the maximum depth for certificate chain verification.

- **Use Session Persistence:** Optional. Check this box to enable persistent sessions from a single logical client to a single backend web server. The following settings configure session persistence. See Session Persistence for more information.
  - **Cookie Name:** The cookie name used to enable session persistence. Specify '*' to match any cookie name.
  - **Disable Fallback:** Check this box to disable fallback when the original server is unavailable.
- **Health Check:** Required. Specify the test parameters to confirm the health of backend servers.
  - **Protocol:** Required. Specify the protocol to use, either HTTP or TCP.

    Important

    Configure your health check protocol to match your application or service.
  - **Port:** Required. Specify the backend server port against which to run the health check.

    Tip

    You can enter the value '0' to have the health check use the backend server's traffic port.
  - **Interval in ms:** Optional. Specify how frequently to run the health check, in milliseconds. The default is 10000 (10 seconds).

- **Timeout in ms:** Optional. Specify the maximum time in milliseconds to wait for a reply to a health check. A health check is successful only if a reply returns within this timeout period. The default is 3000 (3 seconds).
- **Number of retries:** Optional. Specify the number of retries to attempt before a backend server is considered "unhealthy". The default is '3'.
- **URL Path (URI):** (HTTP only) Required. Specify a URL endpoint against which to run the health check.
- **Status Code:** (HTTP only) Optional. Specify the status code a healthy backend server must return.
- **Response Body Regex:** (HTTP only) Optional. Provide a regular expression for parsing the response body from the backend server.

5. Click **Create**.

After your backend set is provisioned, you must specify backend servers for the set. See Managing Backend Servers for more information.

More information:

- Securing Networking: VCN, Load Balancers, and DNS
- Managing Backend Sets
- Editing Health Check Policies
- Managing a Load Balancer

Load balancer SSL certificate expires in *X* days

Issue: A load balancer's SSL certificate expires soon. When the certificate expires, data traffic can be interrupted and security compromised.

Basics:

- To ensure continuous security and usability, SSL certificates must be rotated on a timely basis.
- You can configure an exception in Oracle CASB Cloud Service to reduce alerts from exempted load balancers.

To rotate a load balancer's certificate bundle:

The following procedure is for the Oracle Cloud Infrastructure Console.

To ensure consistent service, you must update (rotate) expiring certificates:

1. Update your client or backend server to work with a new certificate bundle.



Note

The steps to update your client or backend server are unique to your system.

2. Upload the new SSL certificate bundle to the load balancer
    a. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
    b. Click the name of the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
    c. Click the load balancer you want to configure.
    d. In the **Resources** menu, click **Certificates**, and then click **Add Certificate**.
    e. In the **Add Certificate** dialog box, enter the following:
        - **Certificate Name:** Required. Specify a friendly name for the certificate bundle. It must be unique within the load balancer, and it cannot be changed in the Console. (It can be changed using the API.) Avoid entering confidential information.

- **Certificate:** Optional. (Required for SSL termination.) Paste the certificate, in PEM format, in this field.
- **CA Certificate:** Optional. (Recommended for backend SSL termination configurations.) Paste the Certificate Authority certificate (root CA certificate) in this field.
- **Private Key:** Optional. (Required for SSL termination.) Paste the private key for the certificate in this field.
- **Passphrase:** Optional. Specify a passphrase.

f. Click **Add Certificate**.

3. Edit listeners or backend sets (as needed) so they use the new certificate bundle

EDITING A LISTENER:

a. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.

b. Choose the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.

c. In the **Resources** menu, click **Listeners** (if necessary).

d. For the listener you want to edit, click the Actions icon (three dots), and then click **Edit Listener**.

e. In the **Certificate Name** drop-down list, choose the new certificate bundle.

f. Click **Submit**.

EDITING A BACKEND SET:



Warning

Updating the backend set temporarily interrupts traffic and can drop active connections.

a. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.
b. Click the name of the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
c. In the **Resources** menu, click **Backend Sets** (if necessary).
d. Click the name of the backend set you want to edit.
e. Click **Edit Backend Set**.
f. In the **Certificate Name** drop-down list, choose the new certificate bundle.
g. Click **Submit**.

4. (Optional) Remove the expiring SSL certificate bundle



Important

You cannot delete an SSL certificate bundle that is associated with a listener or backend set. Remove the bundle from any additional listeners or backend sets before deleting.

a. Open the navigation menu. Under the **Core Infrastructure** group, go to **Networking** and click **Load Balancers**.

b. Click the name of the **Compartment** that contains the load balancer you want to modify, and then click the load balancer's name.
c. Click the load balancer you want to configure.
d. In the **Resources** menu, click **Certificates**.
e. For the certificate you want to delete, click the Actions icon (three dots), and then click **Delete**.
f. Confirm when prompted.

More information:

- Securing Networking: VCN, Load Balancers, and DNS
- Managing SSL Certificates
- Managing Load Balancer Listeners
- Managing Backend Sets
- Managing a Load Balancer

Object Storage

Public buckets detected

Issue: Public buckets were detected in your tenancy. Confirm that the creation of each public bucket is intentional and authorized. If the bucket is not sanctioned for public access, follow the procedure for changing the visibility of a bucket and make the bucket private.

Basics:

- Carefully assess the business requirement for public access to a bucket. When you enable anonymous access to a bucket, users can obtain object metadata, download bucket objects, and optionally list bucket contents.
- Changing the type of access is bi-directional. You can change a bucket's access from public to private or from private to public.

- Changing the type of access doesn't affect existing pre-authenticated requests. Existing pre-authenticated requests still work.

To change the visibility of a bucket (private or public):

The following procedure is for the Oracle Cloud Infrastructure Console.

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
   A list of the buckets in the compartment you're viewing is displayed. If you don't see the one you're looking for, verify that you're viewing the correct compartment (select from the list on the left side of the page).
2. Click the bucket name to see the bucket details.
   Visibility: shows the current bucket setting, which is **Private** by default.
3. Click **Edit Visibility**.
4. In the **Edit Visibility** dialog box, edit the visibility settings:
   - **Visibility**
     - **Public**
     - **Private**
   - If you select **Public** to enable public access, decide whether or not you want to let users list the bucket contents. Click **Allow users to list objects from this bucket** to set the visibility of bucket object lists.
5. Click **Save Changes**.

More information:

- Securing Object Storage
- Managing Buckets
- Using Pre-Authenticated Requests

Oracle Cloud Security Response to Intel L1TF Vulnerabilities

Intel disclosed a new set of speculative execution side-channel processor vulnerabilities affecting their processors. For more information, see Vulnerability Note VU#584653. These L1 Terminal Fault (L1TF) vulnerabilities affect a number of Intel processors, and they have received the following CVE identifiers:

- CVE-2018-3615, which impacts Intel Software Guard Extensions (SGX) and has a CVSS Base Score of 7.9.
- CVE-2018-3620, which impacts operating systems and System Management Mode (SMM) running on Intel processors and has a CVSS Base Score of 7.1.
- CVE-2018-3646, which impacts virtualization software and Virtual Machine Monitors (VMM) running on Intel processors and has a CVSS Base Score of 7.1.

See Intel Processor L1TF vulnerabilities: CVE-2018-3615, CVE-2018-3620, CVE-2018-3646 for more information.

Oracle Cloud Infrastructure

Oracle has deployed technical mitigations across Oracle Cloud Infrastructure systems designed to prevent a malicious attacker's virtual machine (VM) instance from accessing data from other VM instances.

However, vulnerability CVE-2018-3620 could enable a rogue user mode process to read privileged kernel memory within the same virtual machine. As a result, if you manage your own operating systems (OS), you are advised to keep up with OS security patches to address this vulnerability.

The following sections contain the details of mitigations and actions.

Oracle Cloud Infrastructure Compute

For details and required actions related to the Compute service's VM and bare metal instances, see Oracle Cloud Infrastructure Customer Advisory for L1TF Impact on the

Compute Service.

Oracle Cloud Infrastructure Database

If you use Autonomous Data Warehouse and Autonomous Transaction Processing, you have no further action to take.

For details and required actions related to Oracle Cloud Infrastructure offerings for VM DB systems, bare metal DB systems, and Exadata DB systems, see Oracle Cloud Infrastructure Customer Advisory for L1TF Impact on the Database Service.

Platform Service and Kubernetes Services on Oracle Cloud Infrastructure

Oracle has deployed technical mitigations designed to prevent malicious attacker's VM instance from accessing data from other VM instances on the same hypervisor.

However, vulnerability CVE-2018-3620 could enable a rogue user-mode process to read privileged kernel memory within the same virtual machine. As a result, Platform Service hosts managed by Oracle are being patched by Oracle. If you manage your own operating systems you're advised to keep up with the OS security patches to address this vulnerability.

Other Oracle Cloud Infrastructure Services

Mitigations designed to protect all other Oracle Cloud Infrastructure services have been deployed. Oracle will notify and coordinate directly with customers for any additional required maintenance activities.

Oracle Cloud Infrastructure Classic and Oracle Platform Service on Oracle Cloud Infrastructure Classic

For more information see Oracle Cloud Infrastructure Classic.

Oracle is deploying technical mitigations designed for Infrastructure and Platform Services on Oracle Cloud Infrastructure Classic. Some customers may experience reboots or downtime associated while deploying these mitigations.

Vulnerability CVE-2018-3620 could enable a rogue user-mode process to read privileged kernel memory within the same virtual machine. As a result, Platform Service hosts managed by Oracle are being patched by Oracle. If you manage your own operating systems you're advised to keep up with the OS security patches to address this vulnerability.

Oracle Cloud Infrastructure Customer Advisory for L1TF Impact on the Compute Service

Intel disclosed a new set of speculative execution side-channel processor vulnerabilities affecting their processors. For more information, see Vulnerability Note VU#584653. These L1 Terminal Fault (L1TF) vulnerabilities affect a number of Intel processors, and they have received the following CVE identifiers:

- CVE-2018-3615, which impacts Intel Software Guard Extensions (SGX) and has a CVSS Base Score of 7.9.
- CVE-2018-3620, which impacts operating systems and System Management Mode (SMM) running on Intel processors and has a CVSS Base Score of 7.1.
- CVE-2018-3646, which impacts virtualization software and Virtual Machine Monitors (VMM) running on Intel processors and has a CVSS Base Score of 7.1.

See the Oracle Cloud Security Response to Intel L1TF Vulnerabilities for more information.

Oracle has deployed technical mitigations across Oracle Cloud Infrastructure systems designed to prevent a malicious attacker’s virtual machine (VM) instance from accessing data from other VM instances.

You should be aware that the vulnerability CVE-2018-3620 could enable a rogue user-mode process to read privileged kernel memory within the same operating system (OS). As a result, you are advised to keep up with OS security patches to address this vulnerability. See Protecting your Compute Instance Against the L1TF Vulnerability for instructions to patch the OS on the instances you manage.

Additional Guidance for Oracle Cloud Infrastructure Bare Metal Instances

Bare metal instances in Oracle Cloud Infrastructure offer you full control of a physical server. Oracle Cloud Infrastructure's network virtualization is designed and configured to protect these instances from unauthorized access of other instances on the Oracle Cloud Infrastructure network, including other customer instances, both VM instances and other bare metal instances.

If you're running your own virtualization stack or hypervisors on bare metal instances, the L1TF vulnerability allows a virtual machine to access privileged information from the underlying hypervisor or other VMs on the same bare metal instance. You should review the Intel recommendations about vulnerabilities CVE-2018-3615, CVE-2018-3620, and CVE-2018-3646, and make changes to your configurations as you deem appropriate.

Protecting your Compute Instance Against the L1TF Vulnerability

Intel disclosed a new set of speculative execution side-channel processor vulnerabilities affecting their processors, for more information, see Vulnerability Note VU#584653. These L1 Terminal Fault (L1TF) vulnerabilities affect a number of Intel processors, and they have received the following CVE identifiers:

- CVE-2018-3615 which impacts Intel Software Guard Extensions (SGX) and has a CVSS Base Score of 7.9.
- CVE-2018-3620 which impacts operating systems and System Management Mode (SMM) running on Intel processors and has a CVSS Base Score of 7.1.
- CVE-2018-3646 which impacts virtualization software and Virtual Machine Monitors (VMM) running on Intel processors and has a CVSS Base Score of 7.1.

See the Oracle Cloud Security Response to Intel L1TF Vulnerabilities for more information.

Recommended Action

Oracle recommends that you patch the operating systems for your existing bare metal and virtual machine (VM) instances, and verify that this includes the patch for the CVE-2018-3620 vulnerability. For VM instances, the Oracle Cloud Infrastructure team has implemented the necessary workarounds designed to mitigate the CVE-2018-3646 vulnerability. For bare metal

instances using virtualization technology, you should also follow these instructions to ensure that they are mitigated against the CVE-2018-3646 vulnerability.

If you're running your own virtualization stack or hypervisors on bare metal instances, you should apply the patch for the CVE-2018-3646 vulnerability.

The information in the following sections detail the commands needed to update your running instances created from Oracle-Provided Images.

The following Oracle-provided image releases have been updated with the recommended patches, so instances created from these images or new image releases include the recommended patches for the L1TF vulnerability.

Oracle-provided images updated with recommended patches for the L1TF vulnerability

- Oracle-Linux-7.5-2018.08.14-0
- Oracle-Linux-7.5-Gen2-GPU-2018.08.14-0
- Oracle-Linux-6.10-2018.08.14-0
- CentOS-7-2018.08.15-0
- CentOS-6.10-2018.08.15-0
- Canonical-Ubuntu-18.04-2018.08.15-0
- Canonical-Ubuntu-16.04-2018.08.15-0
- Canonical-Ubuntu-16.04-Gen2-GPU-2018.08.15-0
- Canonical-Ubuntu-14.04-2018.08.15-0
- Windows-Server-2016-Standard-Edition-VM-Gen2-2018.08.16-0
- Windows-Server-2016-Datacenter-Edition-BM-Gen2-2018.08.16-0
- Windows-Server-2016-Datacenter-Edition-BM-Gen2-DenseIO-2018.08.16-0
- Windows-Server-2012-R2-Standard-Edition-VM-2018.08.14-0
- Windows-Server-2012-R2-Standard-Edition-VM-Gen2-2018.08.14-0
- Windows-Server-2012-R2-Datacenter-Edition-BM-2018.08.15-0

- Windows-Server-2012-R2-Datacenter-Edition-BM-Gen2-DenseIO-2018.08.15-0
- Windows-Server-2012-R2-Datacenter-Edition-BM-Gen2-2018.08.15-0
- Windows-Server-2008-R2-Enterprise-Edition-VM-2018.08.15-0
- Windows-Server-2008-R2-Enterprise-Edition-VM-Gen2-2018.08.15-0

For your running instances created from imported custom images, refer to the operating system (OS) vendor's guidance to patch the OS for the L1TF vulnerability.

PATCHING ORACLE LINUX INSTANCES

For Oracle Linux, the patches for the CVE-2018-3620 and CVE-2018-3646 vulnerabilities are addressed by the same set of patches.

Bare metal instances must have the latest microcode updates from Intel. This step is not required for VM instances.

To install the latest microcode updates, run the following command:

```
# sudo yum update microcode_ctl
```

The microcode RPM should be greater than or equal to `microcode_ctl-2.1-29.2.0.4.el7_5.x86_64.rpm`. This is the version of the microcode package that shipped for the Spectre v3a and Spectre v4 updates. No additional update is required. In addition to the microcode update, you should also patch your bare metal instances using the following set of instructions.

To patch the OS for bare metal and VM instances with downtime

The yum-plugin-security package allows you to use yum to obtain a list of all of the errata that are available for your system, including security updates. You can also use Oracle Enterprise Manager 12c Cloud Control or management tools such as Katello, Pulp, Red Hat Satellite, Spacewalk, and SUSE Manager to extract and display information about errata.

1. To install the `yum-plugin-security` package, run the following command:

   ```
   # sudo yum install yum-plugin-security
   ```

2. Use the `--cve` option to display the errata that correspond to a specified CVE, and to install those required packages, by running the following commands:

```
# sudo yum updateinfo list --cve CVE-2018-3620
# sudo yum update --cve CVE-2018-3620
```

   A system reboot will be required once the package is applied. By default, the boot manager will automatically enable the most recent kernel version. For more information on using `yum update`, visit Installing and Using the Yum Security Plugin.

3. After the system reboots, ensure that the following file is populated.

```
cat /sys/devices/system/cpu/vulnerabilities/l1tf
```

PATCHING WINDOWS INSTANCES

PROTECTING NEW WINDOWS VM AND BARE METAL INSTANCES

When you create a new VM or bare metal instance based on the latest Oracle-provided Windows images, the image includes the Microsoft-recommended patches to protect against the L1TF vulnerability. Windows bare metal instances also include the latest microcode updates from Intel.

There is no further action required from you to protect your new Windows-based VM or bare metal instances from the L1TF vulnerability. You should ensure that you keep the your instances updated with the latest patches as recommended by your OS vendor.

PROTECTING EXISTING WINDOWS VM AND BARE METAL INSTANCES

To update the microcode for existing bare metal instances

Bare metal instances launched before the Oracle-provided Windows images were updated must have the latest microcode updates from Intel. You need to recycle your Windows bare metal instances in order to receive the latest Intel microcode update. This step is not required for VM instances.

1. Create a new custom image of your Windows bare metal instance, see Creating Windows Custom Images for more information.

2. Terminate your existing Windows bare metal instance.
3. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Custom Images**. Find the custom image you want to use.
4. Click the Actions icon (three dots), and then click **Create Instance**.
5. Provide additional launch options as described in Creating an Instance.

Once you have completed these steps, perform the steps in the next procedure to update the instance with the latest OS updates from Microsoft.

To patch the OS for bare metal and VM instances with downtime

Windows images include the Windows Update utility, which you can run to get the latest Windows updates from Microsoft. You have to configure the security list on the subnet on which the instance is running to allow instances to access Windows update servers. See Windows OS Updates for Windows Images and Security Lists for more information.

1. Verify that you have installed the latest Windows OS security update from Microsoft.
    a. If automatic updates are turned on, the updates should be automatically delivered to the instance.
    b. To manually check for the latest update, select **Start**.
    c. In **Settings** select **Updates & security** and then select **Windows Update**.
    d. In **Windows Update**, click **Check for updates**.
    e. When you turn on automatic updates, this update will be downloaded and installed automatically. For more information about how to turn on automatic updates, see Windows Update: FAQ.

For additional details see Windows Server guidance to protect against L1 terminal fault.

PATCHING UBUNTU OR CENTOS INSTANCES

When you create a new VM or bare metal instance based on the latest Oracle-provided Ubuntu or CentOS images, the image includes the recommended patches to protect against the L1TF

vulnerability, see for more information L1 Terminal Fault (L1TF) and L1TF - L1 Terminal Fault Attack - CVE-2018-3620 & CVE-2018-3646.

For existing VM or bare metal instances you should follow the guidance provided by the OS vendor for patching systems.

Oracle Cloud Infrastructure Customer Advisory for L1TF Impact on the Database Service

Intel disclosed a new set of speculative execution side-channel processor vulnerabilities affecting their processors. For more information, see Vulnerability Note VU#584653. These L1 Terminal Fault (L1TF) vulnerabilities affect a number of Intel processors, and they have received the following CVE identifiers:

- CVE-2018-3615, which impacts Intel Software Guard Extensions (SGX) and has a CVSS Base Score of 7.9.
- CVE-2018-3620, which impacts operating systems and System Management Mode (SMM) running on Intel processors and has a CVSS Base Score of 7.1.
- CVE-2018-3646, which impacts virtualization software and Virtual Machine Monitors (VMM) running on Intel processors and has a CVSS Base Score of 7.1.

See the Oracle Cloud Security Response to Intel L1TF Vulnerabilities for more information.

Oracle has deployed technical mitigations across Oracle Cloud Infrastructure systems designed to prevent a malicious attacker's virtual machine (VM) instance from accessing data from other VM instances.

Autonomous Data Warehouse and Autonomous Transaction Processing

Autonomous Data Warehouse provides fully managed databases optimized for running data warehouse workloads. Autonomous Transaction Processing provides fully managed databases optimized for running online transaction processing and mixed database workloads. Autonomous Data Warehouse and Autonomous Transaction Processing are not affected by the L1TF vulnerabilities, CVE-2018-3615, CVE-2018-3620, and CVE-2018-3646. No further action is required by customers.

Guidance for the DatabaseService on Bare Metal Instances

The Database service on Oracle Cloud Infrastructure bare metal instances offer customers full control over their Oracle Database running on a physical server. Oracle Cloud Infrastructure's network virtualization is designed and configured to protect these instances from unauthorized access from other instances on the Oracle Cloud Infrastructure network, including other customer instances, both VM instances and other bare metal instances.

Actions for Customers with VM DB Systems, Bare Metal DB Systems, or Exadata DB Systems

Vulnerability CVE-2018-3620 could enable a rogue user-mode process to read privileged kernel memory within the same operating system. As a result, you need to patch these systems once these patches are available. These patches will be available shortly and Oracle will update this page when the operating system (OS) patches are published. Oracle will update the Database base images with the latest patches for new instance launches.

Once the patches are available, use the following instructions to patch a running instance:

- For DB systems on bare metal instances, apply the OS patches following the instructions in Updating a DB System.
- For DB systems on a VM instance, configured using the Oracle Cloud Infrastructure Database service, apply the OS patches following the instructions in Updating a DB System.
- For the DB systems on a VM instance configured using the Oracle Platform Service Manager, apply the OS patches following the instructions in Applying Linux OS Security Patches by Using the dbaascli Utility.
- For Exadata DB systems, apply the OS patches following the instructions in Updating an Exadata DB System.

Oracle Cloud Security Response to Intel Microarchitectural Data Sampling (MDS) Vulnerabilities

Intel disclosed four new speculative execution side-channel processor vulnerabilities affecting Intel processors. These vulnerabilities have received the following CVE identifiers:

- CVE-2019-11091: Microarchitectural Data Sampling Uncacheable Memory (MDSUM)
- CVE-2018-12126: Microarchitectural Store Buffer Data Sampling (MSBDS)
- CVE-2018-12127: Microarchitectural Load Port Data Sampling (MLPDS)
- CVE-2018-12130: Microarchitectural Fill Buffer Data Sampling (MFBDS)

For more information, see https://blogs.oracle.com/security/intelmds.

Oracle Cloud Infrastructure

Oracle has deployed technical mitigations across Oracle Cloud Infrastructure systems designed to prevent a malicious attacker's virtual machine (VM) instance from accessing data from other VM instances.

However, if you manage your own operating systems (OS), you are advised to keep up with OS security patches to address this vulnerability.

The following sections contain the details of mitigations and actions.

Oracle Cloud Infrastructure Compute

For details and required actions related to the Compute service's VM and bare metal instances, see Oracle Cloud Infrastructure Customer Advisory for MDS Impact on the Compute Service.

Oracle Cloud Infrastructure Database

If you use Autonomous Data Warehouse and Autonomous Transaction Processing, you have no further action to take.

For details and required actions related to offerings for VM DB systems, bare metal DB systems, and Exadata DB systems, see Oracle Cloud Infrastructure Customer Advisory for MDS Impact on the Database Service.

Oracle Cloud Infrastructure Container Engine for Kubernetes

To help secure your existing worker nodes for the Oracle Cloud Infrastructure Container Engine for Kubernetes Oracle recommends replacing your current node pools with new node pools. Please follow the instructions described in 'Upgrading' the version of Kubernetes running on worker nodes by creating a new node pool. All worker nodes created or upgraded after May 14th, 2019 are not impacted by this security issue.

Other Oracle Cloud Infrastructure Services

Technical mitigations designed to protect all other Oracle Cloud Infrastructure services against the MDS processor vulnerabilities have been deployed. Oracle will notify customers if additional maintenance activities are required.

Oracle Cloud Infrastructure Classic and Oracle Platform Service on Oracle Cloud Infrastructure Classic

For more information see Oracle Cloud Infrastructure Classic.

In response to the MDS processor vulnerabilities, Oracle is performing mandatory maintenance for Infrastructure and Platform Services on Oracle Cloud Infrastructure Classic.

Platform Service hosts managed by Oracle are being patched by Oracle. If you manage your own operating systems, you are advised to keep up with the appropriate OS security patches to address these vulnerabilities.

Oracle Cloud Infrastructure Customer Advisory for MDS Impact on the Compute Service

Intel disclosed four new speculative execution side-channel processor vulnerabilities affecting Intel processors. These vulnerabilities have received the following CVE identifiers:

- CVE-2019-11091: Microarchitectural Data Sampling Uncacheable Memory (MDSUM)
- CVE-2018-12126: Microarchitectural Store Buffer Data Sampling (MSBDS)

- CVE-2018-12127: Microarchitectural Load Port Data Sampling (MLPDS)
- CVE-2018-12130: Microarchitectural Fill Buffer Data Sampling (MFBDS)

For more information, see https://blogs.oracle.com/security/intelmds.

Oracle has deployed technical mitigations across Oracle Cloud Infrastructure systems designed to prevent a malicious attacker's virtual machine (VM) instance from accessing data from other VM instances.

You are advised to keep up with OS security patches to address this vulnerability. See Oracle Cloud Infrastructure Compute Content Impact for instructions to patch the OS on the instances you manage.

Additional Guidance for Oracle Cloud Infrastructure Bare Metal Instances

Bare metal instances in Oracle Cloud Infrastructure offer customers full control of a physical server. Oracle Cloud Infrastructure's network virtualization is designed and configured to protect these instances from unauthorized access of other instances on the Oracle Cloud Infrastructure network, including other customer instances, both VM instances and other bare metal instances

However, for customers running their own virtualization stack on bare metal instances, the MDS vulnerabilities could allow a virtual machine to access privileged information from the underlying hypervisor or other VMs on the same bare metal instance. These customers should review Intel's recommendations about these MDS vulnerabilities and make the recommended changes to their configurations, https://www.intel.com/content/www/us/en/security-center/advisory/intel-sa-00233.html.

Oracle Cloud Infrastructure Compute Content Impact

Intel disclosed four new speculative execution side-channel processor vulnerabilities affecting Intel processors. These vulnerabilities have received the following CVE identifiers:

- CVE-2019-11091: Microarchitectural Data Sampling Uncacheable Memory (MDSUM)
- CVE-2018-12126: Microarchitectural Store Buffer Data Sampling (MSBDS)

- CVE-2018-12127: Microarchitectural Load Port Data Sampling (MLPDS)
- CVE-2018-12130: Microarchitectural Fill Buffer Data Sampling (MFBDS)

For more information, see https://blogs.oracle.com/security/intelmds.

Recommended Action

Oracle recommends that customers patch the operating systems for their existing bare metal and virtual machine (VM) instances and verify that these OS updates include the patch for the MDS vulnerabilities. For VM instances, the Oracle Cloud Infrastructure team has implemented the necessary workarounds designed to mitigate for the MDS vulnerabilities. For bare metal instances using virtualization technology, you should also follow the following instructions

If you are running your own virtualization stack or hypervisors on bare metal instances, you should apply the appropriate patch required to address the MDS processor vulnerabilities.

The information in the following sections detail the commands needed to update your running instances created with Oracle-Provided Images.

The following Oracle-provided image releases have been updated with the recommended patches, as a result instances created using these images or subsequent images include the recommended patches for the MDS vulnerabilities.

Oracle-provided images updated with recommended patches for the MDS vulnerability

- Oracle-Linux-6.10-2019.05.14-0
- Oracle-Linux-7.6-2019.05.14-0
- Oracle-Linux-7.6-Gen2-GPU-2019.05.14-0
- Windows-Server-2008-R2-Enterprise-Edition-VM-2019.05.14-0
- Windows-Server-2008-R2-Enterprise-Edition-VM-Gen2-2019.05.14-0
- Windows-Server-2012-R2-Standard-Edition-VM-2019.05.15-0
- Windows-Server-2012-R2-Standard-Edition-VM-Gen2-2019.05.14-0
- Windows-Server-2012-R2-Standard-Edition-VM-Gen2-E2-2019.05.15-0

- Windows-Server-2012-R2-Datacenter-Edition-BM-Gen2-2019.05.14-0
- Windows-Server-2012-R2-Datacenter-Edition-BM-Gen2-DenseIO-2019.05.15-0
- Windows-Server-2012-R2-Datacenter-Edition-BM-Gen2-E2-2019.05.14-0
- Windows-Server-2016-Standard-Edition-VM-Gen2-2019.05.14-0
- Windows-Server-2016-Standard-Edition-VM-Gen2-E2-2019.05.14-0
- Windows-Server-2016-Datacenter-Edition-BM-Gen2-2019.05.14-0
- Windows-Server-2016-Datacenter-Edition-BM-Gen2-DenseIO-2019.05.14-0
- Windows-Server-2016-Datacenter-Edition-BM-Gen2-E2-2019.05.15-0
- CentOS-6.10-2019.05.15-0
- CentOS-7-2019.05.16-0
- Canonical-Ubuntu-14.04-2019.05.15-0
- Canonical-Ubuntu-16.04-2019.05.15-0
- Canonical-Ubuntu-16.04-Gen2-GPU-2019.05.15-0
- Canonical-Ubuntu-16.04-Minimal-2019.05.15-0
- Canonical-Ubuntu-18.04-2019.05.15-0
- Canonical-Ubuntu-18.04-Minimal-2019.05.15-0

Customers running instances created from imported third-party images should refer to the operating system (OS) vendor's guidance to patch the OS for the MDS vulnerability.

PATCHING ORACLE LINUX INSTANCES

Oracle has released security patches for Oracle Linux 6, Oracle Linux 7, and Oracle VM Server for X86 products. In addition to the OS patches, customers should run the latest version of the microcode from Intel to mitigate these issues. For both bare metal and VM instances, please install the latest Ksplice via uptrack-upgrade.



Note

See Installing Ksplice Uptrack Within the Oracle Cloud Infrastructure for how to install Ksplice.

For Oracle Linux, the patches for the MDS vulnerabilities are addressed by the same set of patches. For further information please see the following:

- https://linux.oracle.com/cve/CVE-2018-12126.html
- https://linux.oracle.com/cve/CVE-2018-12130.html
- https://linux.oracle.com/cve/CVE-2018-12127.html
- https://linux.oracle.com/cve/CVE-2019-11091.html

Bare metal instances must have the latest microcode updates from Intel. This step is not required for VM instances.

To install the latest microcode updates on bare metal instances, run the following command:

```
# sudo yum update microcode_ctl
```

The required versions of microcode_ctl rpms are:

- **Oracle Linux 7**: microcode_ctl 2.1-47.0.4
- **Oracle Linux 6**: microcode_ctl 1.17-1002

No additional update is required. In addition to the microcode update, you should also patch your bare metal instances using the following set of instructions.

To patch the OS for bare metal and VM instances with downtime

The `yum-plugin-security` package allows you to use yum to obtain a list of all errata that are available for your system, including security updates. You can also use Oracle Enterprise Manager 12c Cloud Control or management tools such as Katello, Pulp, Red Hat Satellite, Spacewalk, and SUSE Manager to extract and display information about errata.

1. To install the `yum-plugin-security` package, run the following command:

   ```
   # sudo yum install yum-plugin-security
   ```

2. Use the `--cve` option to display the errata that correspond to a specified CVE, and to install those required packages, by running the following commands:

   ```
   # sudo yum updateinfo list --cve CVE-####-####
   # sudo yum update --cve CVE-####-####
   ```

   Replace `####-####` in the above commands with the relevant CVE numbers.

3. A system reboot will be required once the package is applied. By default, the boot manager will automatically enable the most recent kernel version. For more information on using yum update, visit Installing and Using the Yum Security Plugin.
4. After the system reboots, ensure that the following file is populated:

   ```
   cat /sys/devices/system/cpu/vulnerabilities/mds
   ```

PATCHING WINDOWS INSTANCES

PROTECTING NEW WINDOWS VM AND BARE METAL INSTANCES

When you create a new VM or bare metal instance based on the latest Oracle-provided Windows images, the image includes the Microsoft-recommended patches to protect against the MDS vulnerability. Windows bare metal instances also include the latest microcode updates from Intel. To apply the MDS patch install the latest Windows Updates and reboot the instance. You should ensure that you keep your instances updated with the latest patches as recommended by your OS vendor.

PROTECTING EXISTING WINDOWS VM AND BARE METAL INSTANCES

To update the microcode for existing bare metal instances

Bare metal instances launched before the Oracle-provided Windows images were updated must have the latest microcode updates from Intel. You need to recycle your Windows bare metal instances in order to receive the latest Intel microcode update. This step is not required for VM instances.

1. Create a new custom image of your Windows bare metal instance, see Creating Windows Custom Images for more information.
2. Terminate your existing Windows bare metal instance.
3. Open the navigation menu. Under **Core Infrastructure**, go to **Compute** and click **Custom Images**. Find the custom image you want to use.
4. Click the Actions icon (three dots), and then click **Create Instance**.
5. Provide additional launch options as described in Creating an Instance.

Once you have completed these steps, perform the steps in the next procedure to update the instance with the latest OS updates from Microsoft

To patch the OS for bare metal and VM instances with downtime

Windows images include the Windows Update utility, which you can run to get the latest Windows updates from Microsoft. You have to configure the security list on the subnet on which the instance is running to allow instances to access Windows update servers. See Windows OS Updates for Windows Images and Security Lists for more information.

1. Verify that you have installed the latest Windows OS security update from Microsoft.
    a. If automatic updates are turned on, the updates should be automatically delivered to the instance.
    b. To manually check for the latest update, select **Start**.
    c. In **Settings** select **Updates & security** and then select **Windows Update**.
    d. In **Windows Update**, click **Check for updates**.
    e. When you turn on automatic updates, this update will be downloaded and installed automatically. For more information about how to turn on automatic updates, see Windows Update: FAQ.

For additional details see Windows Server guidance to protect against speculative execution side-channel vulnerabilities.

PATCHING UBUNTU OR CENTOS INSTANCES

The recommended patches to protect against the MDS vulnerabilities are included when you create a new VM or bare metal instance based on the latest Oracle-provided Ubuntu or CentOS images, see Microarchitectural Data Sampling (MDS) and MDS - Microarchitectural Store Buffer Data - CVE-2018-12130, CVE-2018-12126, CVE-2018-12127, and CVE-2019-11091. For existing VM or bare metal instances you should follow the patching guidance provided by the original OS vendor.



Note

Any images published after May 14th, 2019 listed in the image release notes will include the MDS patches. If using earlier images already launched, follow patching instructions.

Oracle Cloud Infrastructure Customer Advisory for MDS Impact on the Database Service

Intel disclosed 4 new speculative execution side-channel processor vulnerabilities affecting Intel processors. These vulnerabilities have received the following CVE identifiers:

- CVE-2019-11091: Microarchitectural Data Sampling Uncacheable Memory (MDSUM)
- CVE-2018-12126: Microarchitectural Store Buffer Data Sampling (MSBDS)
- CVE-2018-12127: Microarchitectural Load Port Data Sampling (MLPDS)
- CVE-2018-12130: Microarchitectural Fill Buffer Data Sampling (MFBDS)

For more information, see https://blogs.oracle.com/security/intelmds.

Oracle has deployed technical mitigations across Oracle Cloud Infrastructure systems designed to prevent a malicious attacker's virtual machine (VM) instance from accessing data from other VM instances.

Autonomous Data Warehouse and Autonomous Transaction Processing

Autonomous Data Warehouse provides fully managed databases optimized for running data warehouse workloads.

Autonomous Transaction Processing provides fully managed databases optimized for running online transaction processing and mixed database workloads.

Autonomous Data Warehouse and Autonomous Transaction Processing are not affected by MDS vulnerabilities. These services do not run on their own hypervisor and they do not allow for the execution of untrusted code in their services enclave. Customers can execute code within their own instances and each customer instance is isolated from that of another customer. No further customer action is currently required.

Guidance for the DatabaseService on Bare Metal Instances

The Database service on Oracle Cloud Infrastructure bare metal instances offer customers full control over their Oracle Database running on a physical server. Oracle Cloud Infrastructure's network virtualization is designed and configured to protect these instances from unauthorized access from other instances on theOracle Cloud Infrastructure network, including other customer instances, both VM instances and other bare metal instances. As a result, the Database service on bare metal instances are not affected by the MDS vulnerabilities.

Actions for Customers with VM DB Systems, Bare Metal DB Systems, or Exadata DB Systems

Customers are advised to apply available patches at the earliest possible time. Use the following instructions to patch a running instance:

- For DB systems on bare metal instances, apply the OS patches following the instructions in Updating a DB System.
- For DB systems on a VM instance, configured using the Oracle Cloud Infrastructure Database service, apply the OS patches following the instructions in Updating a DB System.
- For the DB systems on a VM instance configured using the Oracle Platform Service

Manager, apply the OS patches following the instructions in Applying Linux OS Security Patches by Using the dbaascli Utility.

- For Exadata DB systems, apply the OS patches following the instructions in Updating an Exadata DB System.

CHAPTER 25 Storage Gateway

This chapter explains how to use Storage Gateway to connect your on-premise applications with Oracle Cloud Infrastructure.

Overview of Storage Gateway

Storage Gateway is a cloud storage gateway that lets you connect your on-premises applications with Oracle Cloud Infrastructure. Applications that can write data to an NFS target can also write data to the Oracle Cloud Infrastructure Object Storage, without requiring application modification to uptake the REST APIs.



Important

Storage Gateway is the evolution of the Storage Software Appliance that was launched with Oracle Cloud Infrastructure Classic. Now that you're migrating to Oracle Cloud Infrastructure Object Storage, you'll use Storage Gateway, with its enhanced file-to-object transparency and improved scale and performance.

Storage Gateway and Oracle Cloud Infrastructure Concepts

The following concepts are essential to working with Oracle Cloud Infrastructure Storage Gateway.

FILE SYSTEM

A Storage Gateway file system on a local host maps its files and directories to objects with the same names in a corresponding Oracle Cloud Infrastructure Object Storage bucket.

FILE SYSTEM CACHE

Storage Gateway's configurable file system cache enables asynchronous and optimized movement of data to the cloud. The file system cache serves as both a write buffer and a read cache for data storage and retrieval. The write buffer contains data that copied to the disk cache and queued for upload to Oracle Cloud Infrastructure. The read cache contains frequently retrieved data that's accessible locally for read operations.

Proper file system cache configuration is critical to Storage Gateway performance. See Configuring the Cache for File Systems for details.

METADATA

The metadata associated with a Storage Gateway file is stored as custom metadata for the corresponding object in Oracle Cloud Infrastructure Object Storage. Examples of file metadata include object id, creation date, modification date, size, and permissions. Storage Gateway caches all metadata for the file system locally.

NFSv4

NFS is an established and widely adopted distributed file system protocol for handling network storage. NFS lets client computers mount file systems on remote servers and access those remote file systems over the network as though they were local file systems. Storage Gateway performs the NFS to REST API translation needed to interact with Oracle Cloud Infrastructure Object Storage.

ORACLE CLOUD INFRASTRUCTURE

Oracle Cloud Infrastructure is a set of complementary cloud services that lets you build and run a wide range of applications and services in a highly available hosted environment. Oracle Cloud Infrastructure offers high-performance compute capabilities (as physical hardware instances) and storage capacity in a flexible overlay virtual network that is securely accessible from your on-premises network.

TENANCY

A tenancy is a secure and isolated partition within Oracle Cloud Infrastructure where you can create, organize, and administer your cloud resources.

Oracle Cloud Infrastructure Object Storage and Archive Storage

Oracle Cloud Infrastructure offers two distinct storage tiers for you to store your unstructured data. Use the Object Storage **Standard** tier for data to which you need fast, immediate, and frequent access. Use the Archive Storage service's **Archive** tier for data that you access infrequently, but which must be preserved for long periods of time. Both storage tiers use the same manageable resources (for example, objects and buckets). The difference is that when you upload a file to Archive Storage, the object is immediately archived. Before you can access an archived object, you must first restore the object to the Standard tier.



Note

In the Storage Gateway documentation, generic references to Object Storage encompass both the Standard and Archive storage tiers.

Both storage tiers are simple to use, perform well, and scale to an unlimited capacity.

BUCKET

An Object Storage bucket is a logical container for storing objects. A file system created in Storage Gateway maps to a corresponding bucket by the same name in Object Storage. A bucket is associated with a single Oracle Cloud Infrastructure compartment. The compartment has policies that determine what actions a user can perform on the bucket the objects it contains.

OBJECT

An individual file or directory written to a Storage Gateway file system on an NFS share creates an identically named object in the target Object Storage bucket. An object is composed of the object itself and metadata about the object.

NAMESPACE

A logical entity that serves as a top-level container for all Oracle Cloud Infrastructure Object Storage buckets and objects. The namespace enables you to control bucket naming within your tenancy. Each tenancy has one unique and uneditable Object Storage namespace that is global, spanning all compartments and regions. Bucket names must be unique within your tenancy.

COMPARTMENT

A collection of Oracle Cloud Infrastructure-related resources. Only users and groups with access permissions explicitly granted by an administrator can access the resources. Compartments help you organize resources and make it easier to control access to those resources. Object Storage automatically creates a root compartment when a compartment is provisioned. An administrator can then create more compartments within the root compartment and add access rules for those compartments. A bucket can exist in only one compartment.

How Storage Gateway Works

Storage Gateway is installed in an Oracle Cloud Infrastructure compute instance or as a Linux Docker instance on one or more hosts in your on-premises data center. Applications store and retrieve objects from Oracle Cloud Infrastructure Object Storage through *file systems* that you create in Storage Gateway.

Storage Gateway exposes an NFS mount point that can be mounted to any host that supports an NFSv4 client. The Storage Gateway mount point maps to an Object Storage bucket.

There is file to object transparency between Storage Gateway and Object Storage:

- A Storage Gateway file system directory on a local host maps to a bucket with an identical name in Oracle Cloud Infrastructure Object Storage.
- Any file written to a Storage Gateway file system is written as an object with the same name in the associated Object Storage bucket. The system stores associated file attributes as object metadata.

- You can access Object Storage objects directly using native APIs, SDKs, third-party tools, the HDFS connector, and the Oracle Cloud Infrastructure CLI and Console. You use the **Refresh** operation in Storage Gateway to ingest any data that was added or modified directly in Object Storage.

Enterprise applications typically work with files in nested directories. Object Storage buckets, and the objects within those buckets, exist in a flat hierarchy. Storage Gateway flattens the directory hierarchy into nested object prefixes in Object Storage. See Interacting With Object Storage for details.

Recommended Uses and Workloads

The following summarizes some of the ways that you can use Storage Gateway.

DATA TRANSFER

Use Storage Gateway to move data from your on-premises host to Oracle Cloud Infrastructure. Storage Gateway is not a replacement for general-purpose network attached storage (NAS), though it behaves similarly to NAS. Use Storage Gateway's integrated Cloud Sync feature to transfer and synchronize data to Oracle Cloud Infrastructure.

CLOUD TIERING

Use Storage Gateway to expand the capacity of on-premises storage solutions without capital expenditures. Configuring and connecting a Storage Gateway file system with a large cache to Oracle Cloud Infrastructure Object Storage provides unlimited scale to create a workflow in which files get automatically moved to the cloud and retrieved only on demand. Even though on-demand retrieval is slower than access to local storage, capital expenditures or changes to existing tools and software are not required.

BACKUPS

Use Storage Gateway to move files to Oracle Cloud Infrastructure Archive Storage as a cost-effective backup solution. You can move individual files and compressed or uncompressed ZIP or TAR archives. Storing secondary copies of data is an ideal use case for Storage Gateway.

ARCHIVAL

Storage Gateway is ideal for archive use cases.

DISASTER RECOVERY

Storage Gateway lets traditional applications move data to highly durable object storage. When you need to recover data, create a fresh instance of Storage Gateway to return the data from Object Storage.

Uses and Workloads Not Supported

Storage Gateway does not support the following uses and workloads.

GENERAL-PURPOSE NETWORK STORAGE

Storage Gateway isn't a general-purpose storage filer and must not be used as a replacement for traditional network storage appliances.

FILE SYNC AND SHARE

Though Storage Gateway is an effective data mover, it’s not a replacement for file sync and share services. Evaluate Oracle services like Oracle Document Cloud service if you need file sync and share functionality.

CONTENT COLLABORATION

Storage Gateway does not support multiple Storage Gateway instances simultaneously reading from and writing to a single Object Storage bucket. Do not use Storage Gateway as a tool for distributed teams to collaborate on creating and managing content.

FREQUENTLY MODIFIED FILES

Do not use Storage Gateway if you expect your data to be modified frequently. Each time a file is modified and closed, Storage Gateway creates an updated version and uploads it to Object Storage as a new object. Frequently modified data results in substantial inefficiency, in terms of both bandwidth consumption and capacity utilization.

RENAMING LARGE DIRECTORY TREES

Renaming directories in the Storage Gateway works well for a small directory tree. However, renaming a parent directory with many children can be slow. The service updates the object ID of every corresponding child object in the object store to reflect the new path. If you do start a rename, ensure that the action finishes by monitoring the **Pending Uploads** field in the Storage Gateway user interface.

Security Considerations

ADMIN PASSWORD

Because Storage Gateway administrators can create, modify, and delete file systems, follow these password guidelines:

- Set a strong password.
- Ensure that the password is secure.
- Share passwords with others only on a need-to-know basis.

DOCKER

Storage Gateway runs inside a Docker container for security and isolation. Follow these Docker-related guidelines and recommendations:

- Avoid or minimize Docker instance operations.
- Avoid logging in to the Docker container. If there is a genuine requirement to log in to the Docker container, use extreme caution to avoid service disruption. Do not change the Docker configuration or the Docker instance unless instructed to do so by Oracle support personal.
- Although the NFS protocol controls access to the file system from clients, Storage Gateway file systems are also locally mounted inside the Docker container. To prevent unauthorized access to file system data, ensure that a Docker container is accessible only by an administrator or an authorized user.

- Configure the Docker host to limit user access to the Storage Gateway Docker container.
- Files and directories in a Docker container are also visible in the Docker host - typically file systems and directories that are provisioned in the Docker host and mapped to the container. Set the appropriate ownership and modes to ensure that only an administrator or an authorized user can access these folders. We recommend the following:
    - A dedicated Storage Gateway host.
    - Limit who can access the Storage Gateway host.
    - Set firewall rules to limit access to the Docker host and Docker container.
    - Implement backup and retention policies for the files associated with Storage Gateway.

ACCESS CONTROL

Default file system export options are too permissive. Set more restrictive export options so that only trusted NFS clients can access the file system data and metadata. Modify the advanced file system settings for **NFS Allowed Hosts** and **NFS Export Options** to restrict access to a file system. In addition to NFS protocol security, you can also set up and configure a firewall on the host to further control access to the file system. UID/GID/modes control access to files and directories. Set the appropriate ownership mode to protect sensitive data.

OBJECT STORAGE

Files in a file system are uploaded to Oracle Cloud Infrastructure and stored as objects in an Object Storage bucket. Associated file attributes are stored as object metadata. Access control for Object Storage is different from access control for a traditional file system. Anyone with permission to read or modify any object in the bucket can read or modify all objects in the bucket. To protect sensitive data, set up Oracle Cloud Infrastructure IAM policies to limit who can access objects in the bucket.

Storage Gateway transfers data to Oracle Cloud Infrastructure using HTTPS, which encrypts data packets in flight between Storage Gateway and the cloud. Data written to Object Storage is always automatically encrypted in the cloud.

Limits on Storage Gateway Resources

See Service Limits for a list of applicable limits and instructions for requesting a limit increase.

Other limits include:

- Ensure that the number of file systems per Storage Gateway doesn't exceed 10. For best performance, host each file system on a dedicated Storage Gateway.
- Ensure that the number of objects stored in a Storage Gateway file system doesn’t exceed 100 million. For datasets that consist of more than 100 million objects, distribute the objects across multiple Storage Gateways.
- Ensure that you configure adequate local storage for file system cache. Storage Gateway warns you if you have configured less than the recommended 500 GB.
- The minimum amount of memory required for any Storage Gateway file system is 16 GB.
    - File systems with up to 50 million files require 32 GB of memory.
    - Large file systems with up to 100 million files require 64 GB of memory.
- The number of files in the cache is limited to 20,000 regardless of the specified cache size in bytes.
- To improve the efficiency of file ingestion, cloud upload operations, and to reduce the number of objects in the namespace, bin-pack or zip small files before writing them to Storage Gateway.

Storage Gateway Release Notes

Release notes provide version-specific release information and important Storage Gateway issues that you need to be aware of:

https://docs.cloud.oracle.com/iaas/releasenotes/services/storage-gateway/

Features of Storage Gateway

This topic highlights key features of Storage Gateway.

POSIX-Compliant NFS Access to Oracle Cloud Infrastructure Object Storage

Using Storage Gateway, your applications can interact with Oracle Cloud Infrastructure Object Storage through standard NFSv4 protocols. You connect Storage Gateway file systems to Object Storage buckets. Storage Gateway stores files as objects in an Oracle Cloud Infrastructure Object Storage bucket and supports multipart uploads for large objects. Object Storage does not, however, support symbolic links, hard links, or special device files.

Data Integrity with Checksum Verification

The built-in data integrity checks ensure that data is validated as it moves through the data path from Storage Gateway to Oracle Cloud Infrastructure Object Storage. Checksum verification helps ensure data integrity. Metadata integrity checks ensure that the metadata is in a consistent state. The checksum for each file can be read using a custom interface.

Large File Support

The Oracle Cloud Infrastructure Object Storage service supports multipart uploads for faster, more efficient, and resilient uploads. Storage Gateway uses multipart upload for files larger than 128 MB. With multipart uploads, individual parts of an object can be uploaded in parallel to reduce the amount of time you spend uploading. Multipart uploads also minimize the impact of network failures by letting you retry a failed part upload instead of requiring you to retry an entire object upload. See Using Multipart Uploads for details.

Support for Data Archival

In addition to uploading to buckets in the Object Storage Standard tier, Storage Gateway supports uploading to and restoring objects from buckets in the Archive Storage tier.

When you create a file system, you specify the storage tier in which to create the corresponding Object Storage bucket.

- The default Standard Object Storage tier is used for storing data to which you need fast, immediate, and frequent access.
- The Archive Storage tier is used for storing data that is accessed infrequently and requires long retention periods.

While Archive Storage is more cost effective than Object Storage for preserving cold data, you must first *restore* the objects before you can access them. The restoration process can take up to four hours depending on the size of the object. See Overview of Archive Storage and Restoring Files and Objects from Archive Storage for details.

Storage Gateway supports Oracle Cloud Infrastructure Object Storage object lifecycle policies to manage the archiving and deletion of objects in a bucket according to a pre-defined schedule. Using object lifecycle policies, you can specify bucket creation in the Standard Object Storage tier, and then create a policy to schedule the subsequent movement of data to the Archive Storage tier. This lifecycle policy archival method is useful if you have on-premises applications that generate intermediary or temporary files and directories that are inappropriate for immediate archival. See Using Object Lifecycle Management for details.

Automated Object Deletion

When you delete a Storage Gateway file from a file system, the corresponding object in Object Storage is automatically deleted.

Quick Access to Select Files with Cache Pinning

Storage Gateway lets you *pin* files to the file system cache for quick access. You can pin files to the cache for file systems connected to either the Object Storage Standard or Archive tier.

When you write a file to your Storage Gateway file system, the file is initially stored in the file system cache, and then asynchronously uploaded to your Oracle Cloud Infrastructure bucket. After a file has been uploaded, the cache manager can remove the file from the file system cache. To meet the cache threshold specified for the file system, cache is reclaimed using the *Least Recently Used* (LRU) cache management policy. If you want specific files to be available

in the cache for quick access, you can pin the files to the file system cache. Once pinned, files are not removed from the file system cache until you explicitly unpin them.

Storage Gateway Health Check

The Storage Gateway performs automated "health checks" on the system to monitor the status of the following:

- Storage Gateway services and resources
- Local storage, file system cache, metadata storage, and log storage

Integrated Cloud Transfer and Synchronization (Cloud Sync)

Storage Gateway 1.2.1 provides an integrated cloud transfer and synchronization feature called Cloud Sync that lets you back up and transfer files on local storage to and from Oracle Cloud Infrastructure Object Storage buckets. This new feature replaces the independent, downloadable cloud sync utility that was available in the previous Storage Gateway version.

You can use the Storage Gateway management console or CLI to create, monitor, and manage Cloud Sync jobs similar to other enterprise NAS backup/replication offerings. Cloud Sync runs as part of the Storage Gateway software inside the Docker instance on the host.

Getting Started With Storage Gateway

This topic provides recommendations for getting started with Storage Gateway.

Recommended Reading

- If you have not done so already, read Overview of Storage Gateway. That topic describes:
    - Key concepts for understanding both Storage Gateway and Object Storage.
    - Important security considerations.

    - Recommended uses and workloads.
    - Uses and workloads to avoid.
- Configuring the cache for file systems is key to Storage Gateway. Read Configuring the Cache for File Systems to understand the importance of the file system cache and the guidelines for configuring the cache when you add a file system.
- To understand the prerequisite tasks and requirements for interacting with Object Storage, read Interacting With Object Storage .

Next Steps for Setting Up Storage Gateway

Key topics for setting up Storage Gateway include:

- Installing Storage Gateway
- Logging In to the Storage Gateway Management Console
- Creating Your First File System
- Mounting File Systems on Clients

Next Steps for Using Storage Gateway

Key topics for using and managing Storage Gateway include:

- Managing File Systems
- Managing Storage Gateway
- Monitoring Storage Gateway

Configuring the Cache for File Systems

Storage Gateway caches frequently retrieved data on the local host, minimizing the number of REST API calls to Oracle Cloud Infrastructure Object Storage and enabling faster data retrieval. You configure the cache for a file system when you create the file system. See Creating Your First File System and Adding a File System.

About File System Cache

The file system cache serves two roles for data storage and retrieval, as both a write buffer and a read cache. The write buffer contains data that has been copied to the disk cache and is queued to be uploaded in your Oracle Cloud Infrastructure tenancy. The read cache contains frequently retrieved data that's accessible locally for read operations.

When an application transfers files through an NFS share, the write buffer can contain many files that are queued and pending upload. If the host on which Storage Gateway is installed fails or Storage Gateway stops abruptly, the pending upload operations persist on the local disk. When Storage Gateway restarts, the pending upload operations resume and the data is uploaded to Oracle Cloud Infrastructure.

When you retrieve data from Oracle Cloud Infrastructure, the data is stored in the Storage Gateway read cache. Read cache allows subsequent I/O operations to that file to be done at local disk speed.

When the read cache is full or reaches the configured limit, Storage Gateway removes files from the cache based on a least recently used (LRU) algorithm. Files pending upload to your tenancy are not removed from cache. You can also preserve files that you do not want removed from cache.

For more information on how to preserve files in the cache, see Preserving Files in the File System Cache.

Configuring Local Storage for File Systems and Cache

Storage Gateway uses local storage attached to the server (or virtual server) for hosting the file systems and cache. Files written to a file system in Storage Gateway are uploaded to the associated Object Storage bucket, with a portion of the file set maintained locally in the file system as a warm cache.

For optimal performance, reliability, and fault tolerance, follow these guidelines when configuring the local Storage Gateway storage:

- Allocate dedicated local storage for each Storage Gateway file system, and associated metadata and logs.
- Multiple disks (hard disk drives or solid-state drives) in a RAID10 set provide an optimal balance of performance, reliability, and fault tolerance. Alternatively, you can use RAID6.



Important

Avoid RAID0 or single disk (no RAID) because of the potential for data loss upon disk failure.

- Provision storage that can accommodate the read cache **and** write buffer (for ingesting new files) without ever becoming more than 80% full.

  In general, use storage that is at least 1.5 times the size of the file set that you want to hold in read cache. For example, assume that the expected size of the entire file set is 50 TB and 10% (5 TB) of that file set is accessed frequently. Ensure that the file system cache storage has at least 7.5 TB of usable capacity. If the cache size reaches a near-full threshold, any data ingestion results in an out of space error in Storage Gateway.
- When you provision local storage at installation time, ensure that you allocate at least the recommended 500 GB for file system cache. Otherwise, Storage Gateway generates a warning message.

Determining File System Cache Size

Remember that the file system cache of Storage Gateway serves two roles, as both a write buffer and a read cache. You can specify the maximum size for the read cache. The write buffer uses any remaining available space in the file system cache. You do not explicitly specify a size for the write buffer.

The maximum size of the write buffer, however, is an important part of determining the cache size. The write buffer size increases when data is ingested in Storage Gateway and decreases

after the data is uploaded to the cloud. The write buffer cannot be removed from the file system cache.



Important

When the write buffer uses all of the available file system cache space, any data ingestion results in an out of space error in Storage Gateway.

Use the following guidelines to determine the appropriate setting for the write buffer:

- Identify the amount of data to be uploaded in Storage Gateway. If a large amount of data is uploaded, the Storage Gateway write buffer can reach its maximum size. Exceeding the write buffer leads to I/O failure as the file system cache has no space available. If you can regulate data ingestion, you can increase the file system cache space and avoid I/O failure. You can regulate I/O by pausing after a certain amount of data is ingested or by periodically allowing uploads to complete before ingesting more data. For example, you can use this approach for backup/cron jobs when the file system cache space is less than the amount of data to be ingested.
- Calculate the amount of data that is ingested on any typical day or week in Storage Gateway. Also, calculate the amount of data that is ingested over a time period, based on the available bandwidth or historical data. Ensure that the difference between these calculations does not exceed the write buffer size.
- Some applications can handle I/O failure, and then resume writing data. In this case, consider setting the cache size to the amount of data that you'd like the application to tolerate before the cache space can be reclaimed.

Configuring the read cache is optional and depends on Storage Gateway workload. While the default setting for read cache is appropriate for most workloads, consider configuring a larger read cache if Storage Gateway must retrieve a significant amount of data from the cloud.

Use the following guidelines to determine the appropriate setting for the read cache:

- The default limit of the read cache size is the lower of 300 GB or the storage volume size.
- Do not set the read cache maximum to the size of the local storage. Doing so allocates 100% of the storage for read cache and does not leave any capacity for ingesting new files. If there is no available space for new file ingestion, Storage Gateway stops ingesting data and begins evicting files from the read cache to create space. Always preserve some space on local storage for ingestion.
- Oracle recommends starting with a read cache setting that is 50% of the size of the local storage, leaving 50% for ingestion. Monitor the available capacity on the local storage over time, especially after periods of high, or sustained, ingestion activity. If the available capacity remains consistently above 30%, consider increasing the read cache size. If the available capacity is consistently below 20%, consider decreasing the read cache size.
- Set the read cache size to equal the amount of data that you anticipate to be accessed frequently, while leaving enough capacity for the write buffer.

After you size the cache, you can configure the read cache when creating the file system or later, after monitoring the workload. See Adding a File System and Changing the Properties of a File System.

Preserving Files in the File System Cache

When you write a file to your file system, the file is initially stored in the file system cache, and then uploaded to your Oracle Cloud Infrastructure tenancy. After a file has been uploaded, the cache manager can remove the file from the file system cache. To meet the cache threshold specified for the file system, cache is reclaimed using the *Least Recently Used* (LRU) cache management policy. If you want specific files to be available in the cache for quick access, you can *pin* the files to the file system cache. Once pinned, files are not removed from the file system cache until you explicitly unpin them. You can view the **Maximum Read Cache Size in GiB** for a selected file system in the management console under **Settings**.

You can pin files connected to both **Standard** and **Archive** storage tiers to file system cache. Files that you write to a file system are always uploaded to your tenancy, regardless of whether the files are pinned to the cache.

If the file that you want to pin to cache is not present in the cache, the file is automatically downloaded to the cache if the file system is connected to a **Standard** storage tier. If that file belongs to a file system connected to an **Archive** storage tier, you must first restore the file before the file can be downloaded to the cache. See Restoring Files and Objects from Archive Storage for details.



Important

- By default, the cache pinning feature is enabled on all file systems.
- When selecting the files for cache pinning, consider the overall cache threshold and calculate the residual cache space that remains available for normal cache operations. For example, assume that your cache threshold is 1 TB and you estimate that the files you want to pin to cache occupy 300 GB. That leaves 700 GB of usable space in your cache after pinning the files.
- When you restore a file from the `Archive` storage tier, the file moves to the `Standard` storage tier. The file remains in `Standard` storage for 24 hours or the retention duration you specify. The continued availability of the file in the cache depends on the LRU operation. However, if you pin such a file to the cache, the restored file remains in the cache until you unpin the file.

Enabling and Managing Cache Pinning

To perform cache pinning operations for a file system, run the following command from the NFS client on which the file system is mounted:

```
cat /path/to/mountpoint/<file_path>:::cache:cache_command[:argument]
```

The following table lists the cache pinning operations and the corresponding command and argument for each operation:

Operation	Cache Command	Argument
Enable cache pinning for a file system. By default, cache pinning is enabled for all file systems.	`set-preserve-option`	`true`
Get the cache pinning status for a file system.	`get-preserve-option`	No argument
Disable cache pinning for a file system.	`set-preserve-option`	`false`
List the files that are pinned to the cache.	`list-preserve`	No argument
Remove deleted files from the preserve list.	`list-preserve-update`	No argument
Add a file to the preserve list.	`add-preserve`	No argument
Remove a file from the preserve list.	`remove-preserve`	No argument
Clear the preserve list.	`clear-preserve`	No argument

Example Commands

- To enable cache pinning for the `myFS` file system:

  ```
  cat /mnt/gateway/myFS/:::cache:set-preserve-option:true
  ```

- To get the cache pinning status for `myFS`:

  ```
  cat /mnt/gateway/myFS/:::cache:get-preserve-option
  ```

  If cache pinning is enabled for the file system, the output of this command is `true`. Otherwise, the output is `false`.

- To disable cache pinning for the `myFS` file system:

  ```
  cat /mnt/gateway/myFS/:::cache:set-preserve-option:false
  ```

- To add a file `myFile` of the `myFS` file system to the preserve list:

  ```
  cat /mnt/gateway/myFS/myFile:::cache:add-preserve
  ```

- To find out which files are added to the preserve list of the `myFS` file system:

  ```
  cat /mnt/gateway/myFS/:::cache:list-preserve
  ```

  Sample output of the preceding command:

  ```
  ["/doNotDelete.txt", "/myFileMetadata", "/myFile"]
  ```

- To remove the file `myFile` from the preserve list

  ```
  cat /mnt/gateway/myFS/myFile:::cache:remove-preserve
  ```

- To update the preserve list when the output of the `cache:list-preserve` command indicates that a pinned file has been removed from the file system:

  ```
  cat /mnt/gateway/myFS/:::cache:list-preserve-update
  ```

  Sample of the original preserve list:

  ```
  ["/doNotDelete.txt", "/myFileMetadata"]
  ```

  Output of the `cache:list-preserve` command after the file `myFileMetadata` is removed from the cache:

  ```
  ["/doNotDelete.txt", "Status: 1 files appear to no longer exist. Please run list-preserve-update"]
  ```

Output of the `cache:list-preserve-update` command:

```
["/doNotDelete.txt"]
```

- To clear the preserve list for a file system:

```
cat /mnt/gateway/myFS/:::cache:clear-preserve
```

Understanding Storage Gateway Performance

This topic covers the performance characteristics of Storage Gateway and the ways you can maximize its efficiency.

Performance Characteristics

It is important to understand the basic performance characteristics of Storage Gateway:

- Because there is transactional overhead for each file, Storage Gateway generally exhibits better performance with large files than with small files. Storage Gateway can only upload data as fast as your connection and the storage host allows. Storage Gateway buffers the data in local disk storage while waiting to upload to Oracle Cloud Infrastructure. Once a file is uploaded, the file's local copy can be removed from the cache to free up space. If file system cache space falls below 10 GB, application I/O and file/directory creation can fail.
- Modifying a file involves uploading a whole new copy of that file. This behavior is not efficient for frequently modified large files.
- Storage Gateway does not support frequently modified files such as logs, databases, or virtual disks. Storage Gateway depends on the closing of a file to trigger an upload of that file. If a file never closes or is modified frequently, the upload event cannot successfully occur.
- Upload throughput to Object Storage (WAN) is typically slower than NFS client throughput (LAN). As a result, Storage Gateway can accumulate a large amount of data that is pending upload.
- You can attach Storage Gateway to an existing Object Storage bucket that contains data. The service is optimized for this type of initialization. Storage Gateway can

initialize about 700 thousand files per hour with a hard disk drive-based cache. It can initialize about 7 million files per hour with an NVMe SSD-based cache.

Factors That Affect Performance

To get the maximum performance benefits from Storage Gateway, follow the practices documented at Best Practices for Using Storage Gateway.

In addition to having sufficient memory and file system cache space, Oracle recommends that you use SSDs to help improve NFS ingestion rate.

Storage Gateway is tuned for maximum upload and download performance by default. No additional tuning is needed.

Performance Testing

While measuring performance is complex and open to variability, we have observed the following in performance benchmark tests with 10-Gb/s link speed:

Upload Speeds

- Single-file upload: 100–120 MB/s (using a 1.3-TB file)
- Multiple-file upload: 450–500 MB/s (using 4–8 files ranging in size from 10 GB to 50 GB)

Download Speeds

- Single-file download: 300–350 MB/s (using a 1.3-TB file)
- Multiple-files download: 700–750 MB/s (using 3 files ranging in size from 50 GB to 200 GB)



Note

Using FastConnect with Storage Gateway makes optimal use of full link speed. We have customers using FastConnect with 10-Gb/s link speed and have seen each gateway achieve 400–450 MB/s uploads to Oracle Cloud Infrastructure.

See FastConnect Overview for more information.

Testing Network Bandwidth

Storage Gateway provides a diagnostic command that you can use to test the bandwidth in your environment and ensure that you get the expected upload and download speeds. The amount of data transferred depends on these factors:

Bandwidth * Delay Product (bits) = total_available_bandwidth (bits/sec) x round_trip_time (sec)



Note

Different buckets can have different upload and download speeds.

The round_trip_time can vary by region.

To run the diagnostic command:

You need `root` permissions to run the `diag` command.

1. Using SSH, log in to the host on which you installed Storage Gateway.
2. Run the `diag` command, specifying the Storage Gateway file system name:

```
[root@ocisg-ashburn opc]# sudo docker exec ocisg cat /mnt/gateway/<file_system_name>/:::diag:oci-network-speed-test
```

The `diag` command responds with the average upload speed, for example:

```
Average Upload Speed = 217 MB/s
```

Interacting With Object Storage

This topic helps you understand the Oracle Cloud Infrastructure Object Storage environment and how it interacts with a Storage Gateway.

Creating the Required IAM Users, Groups, and Policies

An Oracle Cloud Infrastructure administrator must perform prerequisite tasks in preparation for data movement between Storage Gateway and Object Storage. If you are new to Oracle Cloud Infrastructure, we recommend that you read Setting Up Your Tenancy.

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

Access to resources is provided to groups using policies and then inherited by the users that are assigned to those groups. For details on creating groups, see Managing Groups.

For Storage Gateway, an administrator creates these groups with the following policies:

```
Allow group <group_name> to manage buckets in compartment <compartment_name>
```

```
Allow group <group_name> to manage objects in compartment <compartment_name>
```

Content Consistency Between Storage Gateway and Object Storage

Changes to the files in Storage Gateway, including create, write, update, and delete, eventually are consistent with Object Storage. Uploads are asynchronous and buffered for performance, so Storage Gateway file changes might not yet be reflected in Object Storage.

You can access, modify, and upload objects directly to a bucket using Object Storage native APIs, SDKs, the CLI, the Console, or the HDFS connector. Objects modified in these ways do

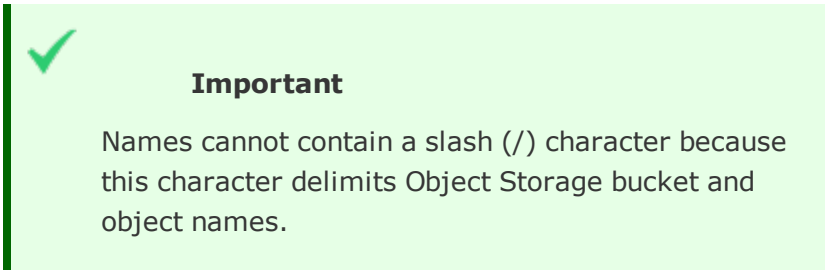
not appear as files in Storage Gateway until you click **Refresh** in the Storage Gateway management console.

Name Restrictions

Storage Gateway file and file system names must adhere to Object Storage bucket and object name restrictions and guidelines.

Use the following guidelines for naming file systems:

- Use from 1 to 256 UTF-8 characters.
- Valid characters are letters (upper or lower case), numbers, hyphens, underscores, and periods.



- Do not include confidential information.
- Make the name unique within a Storage Gateway instance.

Use the following guidelines for naming files:

- Use from 1 to 1024 characters.
- Valid characters are letters (upper or lower case), numbers, and characters other than linefeed, newline, and NULL.
- Use only Unicode characters for which the UTF-8 encoding does not exceed 1024 bytes. Clients are responsible for URL-encoding characters.
- Do not include confidential information.

- Make the name unique within the bucket. Do not use the name of an existing object within the bucket when naming an object unless you intend to overwrite the existing object with the contents of the new or renamed object.

Custom Metadata

POSIX file and directory attributes are stored in custom metadata. These attributes include `uid`, `gid`, `mode`, `atime`, `ctime`, and `mtime`. If existing objects in Object Storage are missing the required custom metadata, Storage Gateway assigns the following default values:

- `uid`=0
- `gid`=0
- `mode`=0644 for file and 0755 for directory

The custom metadata is not updated in Object Storage until a file operation triggers Storage Gateway to update the file in Object Storage.Timestamp metadata (`atime`, `ctime`, and `mtime`) are expressed in milliseconds. Access modes are expressed in octal and include file/directory bit.

The custom metadata names follow these guidelines:

- Only ASCII characters.
- A maximum of 128 bytes.

The custom metadata values follow these guidelines:

- Only UTF-8 characters.
- A maximum of 256 bytes.

Understanding Directory and File Hierarchy Translations in Object Storage

Within an Object Storage namespace, buckets and objects exist in a flat hierarchy. Storage Gateway flattens the file system directory hierarchy into nested object prefixes in Object Storage.

For directories:

- A Storage Gateway file system called `myFS` that contains a directory called `myDir`, appears in Object Storage as:

  ```
  n/<os_namespace>/b/myFS/o/myDir/
  ```

- A Storage Gateway file system called `myFS` that contains a `myDir` subdirectory called `mySubDir`, appears in Object Storage as:

  ```
  n/<os_namespace>/b/myFS/o/myDir/mySubDir/
  ```

You can distinguish a Storage Gateway directory from a Storage Gateway file in the following ways:

- Directories have a trailing slash **/**.
- Directory size or length is **0** (zero).

For files:

- A Storage Gateway file system called `myFS` that contains a directory called `myDir` with a file called `file1`, appears in Object Storage as:

  ```
  n/<os_namespace>/b/myFS/o/myDir/file1
  ```

- A Storage Gateway file system called `myFS` that contains a `myDir` subdirectory called `mySubDir` with a file called `file2`, appears in Object Storage as:

  ```
  n/<os_namespace>/b/myFS/o/myDir/mySubDir/file2
  ```

You can distinguish a Storage Gateway file from a Storage Gateway directory in the following ways:

- Directories have a trailing **/** and files do not.
- File length can be **0** (zero) or non-zero, but directory length is always **0** (zero).

Internal Storage Gateway Objects

Storage Gateway creates some special internal objects in Object Storage. These objects have a `/gateway` directory prefix. For example:

```
/n/<object_storage_namespace>/b/<bucket>/o//gateway
```



Important

Do not modify or remove the objects in the special `/gateway` directory. These objects are critical for Storage Gateway operation.

Installing Storage Gateway

This topic provides instructions for installing the Storage Gateway software.

Prerequisites

These instructions assume that you are familiar with the administration and configuration commands of the operating system on your host machine. To install Storage Gateway, your host system must meet certain hardware and software requirements.

Hardware Recommendations and Requirements

To run Storage Gateway, the host machine must meet the following requirements:

- Two dual-core CPUs or better. Oracle recommends 4-core CPUs.
- Minimum memory requirements:
  - 16 GB for required for any Storage Gateway file system.
  - 32 GB for file systems up to 50 million files.
  - 64 GB for file systems up to 100 million files.
- The recommended local storage disk size is 600 GB, which includes 500 GB for the file system cache, 80 GB for metadata storage, and 20 GB for log storage.



Important

Provision local storage before installing Storage Gateway. For best performance, allocate dedicated local storage file systems for the Storage Gateway cache, the metadata, and the logs. The installation script prompts you for the paths to your Storage Gateway file system cache, metadata storage, and log storage locations. Follow the disk size recommendations provided by the installer.

Oracle recommends that you use the XFS file system for the file system cache, metadata, and logs. XFS is a 64-bit file system designed for parallel I/O. Parallel I/O allows a system to scale based on the number of I/O threads and file system bandwidth.

Software Requirements

- Oracle Linux 7 with UEK Release 4 or later.



Note

If you create an Oracle Cloud Infrastructure Compute instance to host Storage Gateway, the instance creation wizard provides an option to choose the operating system image.

- Docker 1.12.6 or newer. Docker is an open platform for building, shipping and running

distributed applications. For more information, see https://www.docker.com/.

- NFSv4.



Note

The Storage Gateway installation software automatically installs Docker and the NFS protocol.

Hosting Storage Gateway on an Oracle Cloud Infrastructure Compute Instance

To host Storage Gateway on an Oracle Cloud Infrastructure Compute instance, you need:

- An SSH key pair in PEM format.
  - To create a key pair, see Creating a Key Pair.
  - If your public key is not in PEM format, use the following command to convert it:

    ```
    ssh-keygen -f <key_name>.pub -e -m pem
    ```

- An Oracle Cloud Infrastructure user account with an API signing key (the public key from your SSH key pair).
  - If you need to create a user account, see To create a user.
  - To upload an API signing key to an existing user account, see To upload an API signing key.
- A VCN and related resources. For help creating a VCN, see VCNs and Subnets.
  The following configuration points apply to your VCN:
  - Do not select the **Use DNS Hostnames in this VCN** check box unless you plan to use DNS hostnames for your Storage Gateway Compute instance.
  - The security list must include a rule to allow SSL (443) ingress.
  - After you install the Storage Gateway software your host machine, you must add a security list rule to allow communication with the management console port.

More information appears on this page after the Storage Gateway installation instructions.

- A Compute instance. See Creating an Instance.
  The VM.Standard2.4 Compute shape meets the minimum required specifications for Storage Gateway. Large file systems might require an image with more resources.
- A Block Volume. See Creating a Volume.
  - The recommended disk size is 600 GB.
  - Attach the volume to your Compute instance. See Attaching a Volume.
  - If you specify iSCSI as the volume attachment type, you must also connect and mount the volume from the instance for the volume to be usable. For more information, see Volume Attachment Types and Connecting to a Volume.

Installing Storage Gateway

You can install Storage Gateway on an Oracle Cloud Infrastructure Compute instance or an on-premises host that meets the hardware and software requirements.

To install the Storage Gateway software

1. Connect to your Compute instance or on-premises host.
   For help connecting to an Oracle Cloud InfrastructureCompute instance, see Connecting to an Instance.
2. If your host volume is new, you might need to format and mount the disk.



Tip

This task describes the simplest way to create a functional file system to host a Storage Gateway. It uses one device and file system to host the cache, metadata, and log volumes. You specify the paths to those volumes later in this procedure. To optimize performance for your system, you can:

- Create a separate device and file system for each of the cache, metadata, and log volumes.
- Create a single device, but create logical volumes and file systems for the cache, metadata, and log volumes.

To format the disk and create a file system:

a. Run *fdisk*:

```
sudo fdisk /dev/sdb
```

(Optional) Press **m** to view the *fdisk* options.

b. Choose command **g - create a new empty GPT partition table**.

c. Choose command **w - write table to disk and exit**.

d. Run the following command to create an XFS (file system):

```
sudo mkfs -t ext3 /dev/sdb
```

You might see the following warning:

```
/dev/sdb is entire device, not just one partition!
Proceed anyway? (y,n)
```

Choose **y** to proceed.

To mount the formatted volume:

a. Create a mount directory:

```
sudo mkdir /ocisg
```

b. Mount the drive:

```
sudo mount /dev/sdb /ocisg
```

3. Download the Storage Gateway 1.2.1 tar archive.
4. Use the SFTP tool of your choice to copy the tar archive into the `/tmp` folder of the host machine.
5. On the host machine, change directory to `/tmp` and extract the files from the tar archive:

```
cd /tmp
sudo tar xvzf ocisg-1.2.1.tar.gz
```

This command extracts the files from the tar archive into a subdirectory named `ocisg-1.2.1`.

6. Change directory to `ocisg-1.2.1` and run the installation script as `sudo` or `root` user:

```
cd ocisg-1.2.1
sudo ./ocisg-install.sh
```

List of installation script options

Optionally, you can specify the following `ocisg-install.sh` script flags:

- **-a** Runs the installation in advanced configuration mode, which lets you specify ports and the Docker network mode.
  In addition to prompting you for the paths to the metadata storage, cache storage, and log storage, advanced configuration mode also prompts you for:
  - The Docker network mode (`host` or `bridge`).
    Bridge mode is the default. It allows multiple instances of Storage Gateway to run on the same host.

Host mode improves network performance. If you plan to run only one instance of Storage Gateway on the host, Oracle recommends host mode. If you encounter issues with host mode, try bridge mode or contact My Oracle Support.

- The host port to use for the management console.
- The host port to use for NFS access.
- The host port to use for the HTTP REST service.



Tip

For each host port specification, you can designate a port or press **Enter** to let Storage Gateway dynamically allocate the port. You can use the `ocisg configure port` command to change the ports later. See Managing Storage Gateway Using the CLI for details.

- **-d** Installs Storage Gateway at the location you specify instead of the default location of `/opt/ocisg`. For example:

```
sudo ./ocisg-install.sh -d /opt/storagegateway
```

- **-h** Displays the installation script help information.
- **-p** Specifies that Storage Gateway is running behind a proxy server. You can specify multiple proxy arguments. For example:

```
./ocisg-install.sh -p http://myproxy.com:80 -p https://mysecureproxy.com:80
```

- **-q** Runs the installation in quiet mode.

If you supply the paths to the Storage Gateway cache, metadata, and log storage locations using **-m***<path_to_metadata_storage>*, **-c***<path_to_cache_storage>*, and **-l***<path_to_log_storage>*, you are not prompted for input . For example:

```
sudo ./ocisg-install.sh -q -m /ocisg/metadata -c /ocisg/cache -l /ocisg/log
```



Note

Ignore the `devicemapper` warning message if it appears during the installation.

The script guides you through the Storage Gateway installation. Depending on your host machine configuration, some steps can require your input:

a. `Docker does not appear to be installed. Do you want to install docker engine with yum? [y/N]`

Press **y**, and then press **Enter**.

The installation script automatically installs Docker and configures the storage driver for use with Storage Gateway.



Important

If Docker is already installed on your system, the installation script does not automatically configure the storage driver and returns a warning message:

```
Checking that docker is installed and using the
correct version
Found docker version Docker version 18.03.1-ol,
build 0d51d18
The storage appliance requires to set devicemapper
as the docker storage driver.
Please follow the setup link below to enable
devicemapper and rerun the install.
```

Manually verify and update the Docker storage driver to be `devicemapper` as required. See Verifying and Updating the Storage Driver in Docker.

b. `NFS server does not appear to be enabled. Do you want to enable NFS? [y/N]`

   Press **y**, and then press **Enter**.

c. When prompted, press **Enter** accept the default installation location.

d. When prompted, specify the paths to your Storage Gateway cache, metadata, and log storage locations.

   The following examples represent paths for a simple system. Your setup might include paths to separate devices and file systems for each location.

   i. Enter a path for the file system cache. For example:

   ```
   /ocisg/sg/cache
   ```

ii. Enter the path for metadata storage. For example:

```
/ocisg/sg/metadata
```

iii. Enter the path for log storage. For example:

```
/ocisg/sg/log
```

If you receive warnings about cache, metadata, and log storage existing on the same volume, enter **y** to proceed with the installation.

After a successful installation, the script provides the following information:

- The URL to log in to the Storage Gateway management console.
- The NFS port number.
- An example command for mounting your Storage Gateway file systems.

If you installed Storage Gateway on a Compute instance, see Security List Requirements for Compute Instance Installations.

Security List Requirements for Compute Instance Installations

If you installed Storage Gateway on an Oracle Cloud Infrastructure Compute instance, that instance must be able to receive HTTPS connections from other hosts and allow communication with the Storage Gateway management console. To open the necessary port, add an ingress rule to the security list governing the instance's host subnet. To learn about VCN security control, see Security Lists.



Important

This installation task assumes that your existing security list already allows traffic to port 443, as described in the Hosting Storage Gateway on an Oracle Cloud Infrastructure Compute Instance section of this page. If port 443 is not open, you must add a security list rule to open it.

To add a security list rule

1. Open the navigation menu. Under **Core Infrastructure**, go to **Networking** and click **Virtual Cloud Networks**.
2. Click the name of the cloud network (VCN) that hosts your Compute instance.
3. Click **Security Lists**.
4. Click the name of the security list that governs the subnet hosting your Compute instance.
5. Click **Edit All Rules**.
6. Add an ingress rule:
    a. Leave **Stateless** unmarked.
    b. Select **CIDR** for the **Source Type**.
    c. Enter **0.0.0.0/0** to indicate all IP addresses.
    d. Select **TCP** as the **IP Protocol**.
    e. Enter **All** in the **Source Port Range** field.
    f. Specify your Storage Gateway management console port in the **Destination Port Range** field. For example:

```
32769
```

If you do not know the management console port for your Storage Gateway installation, run the following command on the host machine:

```
sudo ocisg info
```

The management console port appears at the end of the management console URL:

```
Management Console: https://exampleCompute:32769
```

7. Click **Save Security List Rules**.
   You can now connect to the compute instance using the public IP address (https:*<public_IP_address>*). See Getting the Instance Public IP Address for details.

Verifying and Updating the Storage Driver in Docker

To verify the storage driver in Docker:

1. Start `docker`:

   ```
   sudo systemctl start docker
   ```

2. Verify the information in `docker`:

   ```
   sudo docker info
   ```

3. Look for `Storage Driver` in the output. For example:

   ```
   Containers: 0
    Running: 0
    Paused: 0
    Stopped: 0
   Images: 0
   Server Version: 18.03.1-ol
   Storage Driver: overlay2
    Backing Filesystem: xfs
    Supports d_type: true
    Native Overlay Diff: false
   Logging Driver: json-file
   Cgroup Driver: cgroupfs
   Plugins:
   ```

```
 Volume: local
 Network: bridge host macvlan null overlay
 Log: awslogs fluentd gcplogs gelf journald json-file logentries splunk syslog
Swarm: inactive
Runtimes: runc
Default Runtime: runc
Init Binary: docker-init
containerd version: 773c489c9c1b21a6d78b5c538cd395416ec50f88
runc version: 4fc53a81fb7c994640722ac585fa9ca548971871
init version: 949e6fa
Security Options:
 seccomp
  Profile: default
 selinux
Kernel Version: 4.1.12-124.15.4.el7uek.x86_64
Operating System: Oracle Linux Server 7.5
OSType: linux
Architecture: x86_64
CPUs: 4
Total Memory: 13.45GiB
Name: ocisg-mahesh
ID: OJ2H:QUSK:BWQZ:25L6:VI5V:CXGX:WFXT:NNNP:RK6O:OS4P:4ABE:JWMV
Docker Root Dir: /var/lib/docker
Debug Mode (client): false
Debug Mode (server): false
Registry: https://index.docker.io/v1/
Labels:
Experimental: false
Insecure Registries:
 127.0.0.0/8
Live Restore Enabled: false

Registries: docker.io (secure)
```



Note

You can ignore the `devicemapper` warning message, if it appears.

If `Storage Driver` is *not* `devicemapper`, do the following:

a. Stop `docker`:

```
sudo systemctl stop docker
```

b. Look for `/etc/docker/daemon.json` in the host.
If the file `daemon.json` does not exist, create it.

c. In the `daemon.json` file, set the `storage-driver` variable to `devicemapper`:

```
{
"storage-driver": "devicemapper"
}
```

d. Restart `docker`:

```
sudo systemctl start docker
```

e. Verify the information in `docker`:

```
sudo docker info
```

Look for `Storage Driver` in the output and verify that the storage driver is `devicemapper`.

Next Step

Logging In to the Storage Gateway Management Console

Logging In to the Storage Gateway Management Console

Use the Storage Gateway management console to create, manage, and monitor file systems.

Storage Gateway provides the URL to access the management console after a successful installation. When you access the management console for the first time, a wizard prompts you to create the administrator credentials and your first file system.



Note

Storage Gateway uses a self-signed certificate for the HTTPS connection. Your browser might warn that the SSL certificate couldn't be verified. If you entered the correct public IP address of the Storage Gateway instance, you can safely ignore this warning. The steps to ignore this warning and go to the management console vary depending on the browser you use.

To log in to the management console:

1. Enter one of the following URLs in a supported web browser:
   - If you installed the software on an on-premises host, enter the URL provided at the end of the Storage Gateway installation script:

     ```
     https://<storagegateway_hostname>:<port_number>
     ```

     For example:

     ```
     https://myStorageGatewayHost:3775
     ```

     Note

     If you cannot access Storage Gateway using the hostname, contact your network administrator. Depending on your network configuration, your Storage Gateway hostname might need to be added to DNS or you might need to use an IP address rather than the hostname.

   - If you installed the software in an Oracle Cloud Infrastructure compute instance, enter the URL as follows:

```
https://<instance_public_IP_address>:<port_number>
```

For example:

```
https://192.168.14.5:3775
```



Note

See Getting the Instance Public IP Address for details.

The console log-in page appears. The page prompts you to set and confirm a password for the Storage Gateway `admin` user.

2. Enter a password that meets the following requirements:
   - From 8 to 32 characters.
   - At least one special character, one numerical character, one uppercase character, and one lowercase character.

Next Step

Creating Your First File System

Creating Your First File System

This topic guides you through creating your first Storage Gateway file system.

Think of a file system as a namespace containing a dataset that's accessible through Storage Gateway. A Storage Gateway file system in this context represents a mapping between a directory on your on-premises host and a bucket in Oracle Cloud Infrastructure Object Storage. When you create a Storage Gateway file system, you define the connection credentials that Storage Gateway uses to connect to your Oracle Cloud Infrastructure tenancy.

When you log in to the management console for the first time, a wizard prompts you to create the administrator credentials and your first file system.

To create your first file system

1. Log in to the management console.
2. Click **File Systems** on the upper-right area of the management console.
3. Click **Create File System**.
4. Enter the required information in **Create a File System**:
    - **File System Name:** A unique, friendly name for the file system. Avoid entering confidential information. Use the following guidelines when naming a file system:
        - Use from 1 to 256 characters.
        - Valid characters are letters (upper or lower case), numbers, hyphens, underscores, and periods. (**Important**: The name cannot contain a slash (/) character because this character delimits bucket and object names in Oracle Cloud Infrastructure Object Storage.)

          Note

          If an Object Storage bucket by this file system name doesn't exist in your tenancy, the bucket is created.

          If a corresponding Object Storage bucket by this file system name exists in your tenancy and there is data in the bucket, Storage Gateway works asynchronously to sync the bucket and file system contents.
    - **Select the Object Storage tier in which you want to store your data**



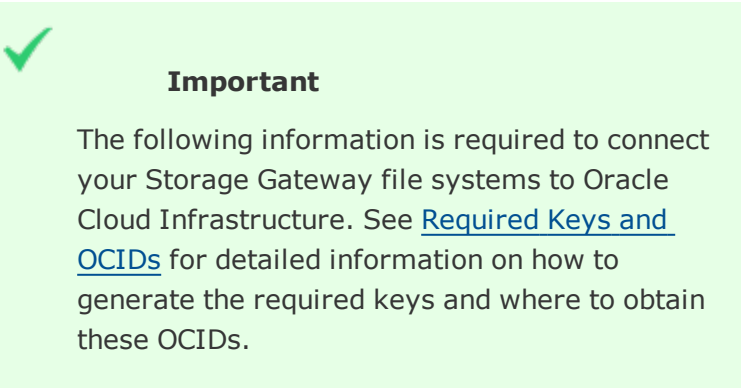
Important

Once set, you cannot change the storage tier in which a bucket resides.

You can use the Oracle Cloud Infrastructure Object Storage object lifecycle policies feature to manage the archiving and deletion of objects in a bucket according to a predefined schedule. See Using Object Lifecycle Management for details.

- **Standard:** The Standard tier is the primary default Object Storage tier for storing data that requires frequent and fast access. See Overview of Object Storage for more information.
- **Archive:** The Archive tier is a special tier for storing data that is accessed infrequently and requires long retention periods. See Overview of Archive Storage for more information. Access to data in the Archive tier is not immediate since you must restore archived data before it's accessible (see Restoring Files and Objects from Archive Storage).

- **Object Storage endpoint:** Required. The Object Storage API endpoint for your service instance. To find the object storage API endpoint for your Oracle Cloud Infrastructure Object Storage tenancy, see the API documentation for Oracle Cloud Infrastructure Object Storage.



- **Compartment OCID:** Required. Unique identifier of your Oracle Cloud Infrastructure Object Storage compartment.
- **Tenant OCID:** Required. Unique identifier of your Oracle Cloud Infrastructure Object Storage tenancy.
- **User OCID:** Required. Unique identifier of your Oracle Cloud Infrastructure Object Storage user.
- **Public Key's Finger Print:** Required. Your Oracle Cloud Infrastructure Object Storage public key fingerprint.
- **Private Key:** Required. Your Oracle Cloud Infrastructure Object Storage private key.
- **Private Key Passphrase:** Required if a passphrase was specified during key creation. Your Oracle Cloud Infrastructure Object Storage private key passphrase.

5. Click **Save**.
   The values you entered must match your Oracle Cloud Infrastructure credentials. If you get an error message, verify your entries, update any incorrect values, and click **Save** again.

6. Click **Show Advanced File System Configuration**.
   Enter the required configuration information or click **Use Default** to accept the default values:
   - **NFS Allowed Hosts:** A comma-separated list of hosts allowed to connect to the NFS export. You can also specify ***** to allow all hosts to connect.
     For example: `2001:db8:9:e54::/64`, `192.0.2.0/24`
   - **NFS Export Options:** The NFS export options.
     Example: `rw`, `sync`, `insecure`, `no_subtree_check`, `no_root_squash`

     Important

     Do not specify the `fsid` option.
   - **Concurrent Uploads:** The number of concurrent uploads to Oracle Cloud Infrastructure.
     This field indicates the maximum number of files that can be concurrently uploaded in Storage Gateway. If the value is 15, the concurrent file uploads can be between 1-15.
     The allowed range is from 1 to 30.
   - **Sync Policy:** The metadata operations are flushed to the disk based on the sync policy, but do not affect on-disk consistency. Currently, only **Posix Standard** is supported. Only the synchronous transactions (like fsync, ODSYNC, and OSYNC) are committed to the disk. All other transactions are handled asynchronously.
   - **Cloud Read-ahead**: The number of blocks to be downloaded and used to *read ahead* when reading files for improved performance.
     Default value: 50
   - **Maximum Read Cache Size in GiB:** The maximum read cache.
     When the read cache is full or reaches the configured limit, Storage Gateway removes files from the cache based on a least recently used (LRU) algorithm.

Files pending upload to your tenancy are not removed from cache. You can also preserve files that you do not want removed from cache.



Note

The number of files in cache is limited to 20,000, regardless of the specified cache size in bytes.

See Configuring the Cache for File Systems for details.

The default value depends on how you provisioned local storage before installing Storage Gateway. The recommended local storage disk size is 600 GB (500 GB for file system cache, 80 GB for metadata, 20 GB for log). If you followed the documented recommendations, the default value for the read cache is approximately 300 GB.

7. Click **Save**.
   The file system is created and appears in the **File Systems** listing.

Next Steps

Connect the file system to a directory on the Storage Gateway host. For more information, see Connecting a File System.

You can also do the following in the management console:

- Set up the NFS export. This directory acts as a mount point. For more information, see Mounting File Systems on Clients.
- Add more file systems. For more information, see Adding a File System.
- View the details of a file system. For more information, see Viewing the Details of a File System.

Managing File Systems

A Storage Gateway file system connects a directory on a local host to an Object Storage bucket in Oracle Cloud Infrastructure. This topic describes how to manage Storage Gateway file systems.

Adding a File System

You can add file systems in Storage Gateway and connect each file system to an Object Storage bucket in your tenancy.

To add a file system

1. Log in to the management console.
2. Click **File Systems** on the upper-right area of the management console.
3. Click **Create File System**.
4. Enter the required information in **Create a File System**:
    - **File System Name:** A unique, friendly name for the file system. Avoid entering confidential information. Use the following guidelines when naming a file system:
        - Use from 1 to 256 characters.
        - Valid characters are letters (upper or lower case), numbers, hyphens, underscores, and periods. (**Important**: The name cannot contain a slash (/) character because this character delimits bucket and object names in Oracle Cloud Infrastructure Object Storage.)



Note

If an Object Storage bucket by this file system name doesn't exist in your tenancy, the bucket is created.

If a corresponding Object Storage bucket by this file system name exists in your tenancy and there is data in the bucket, Storage Gateway works asynchronously to sync the bucket and file system contents.

- **Select the Object Storage tier in which you want to store your data**



Important

Once set, you cannot change the storage tier in which a bucket resides.

You can use the Oracle Cloud Infrastructure Object Storage object lifecycle policies feature to manage the archiving and deletion of objects in a bucket according to a predefined schedule. See Using Object Lifecycle Management for details.

  - **Standard:** The Standard tier is the primary default Object Storage tier for storing data that requires frequent and fast access. See Overview of Object Storage for more information.
  - **Archive:** The Archive tier is a special tier for storing data that is accessed infrequently and requires long retention periods. See Overview of Archive Storage for more information. Access to data in the Archive tier is not

immediate since you must restore archived data before it's accessible (see Restoring Files and Objects from Archive Storage).

- **Object Storage endpoint:** Required. The Object Storage API endpoint for your service instance. To find the object storage API endpoint for your Oracle Cloud Infrastructure Object Storage tenancy, see the API documentation for Oracle Cloud Infrastructure Object Storage.

> **Important**
>
> The following information is required to connect your Storage Gateway file systems to Oracle Cloud Infrastructure. See Required Keys and OCIDs for detailed information on how to generate the required keys and where to obtain these OCIDs.

- **Compartment OCID:** Required. Unique identifier of your Oracle Cloud Infrastructure Object Storage compartment.
- **Tenant OCID:** Required. Unique identifier of your Oracle Cloud Infrastructure Object Storage tenancy.
- **User OCID:** Required. Unique identifier of your Oracle Cloud Infrastructure Object Storage user.
- **Public Key's Finger Print:** Required. Your Oracle Cloud Infrastructure Object Storage public key fingerprint.
- **Private Key:** Required. Your Oracle Cloud Infrastructure Object Storage private key.
- **Private Key Passphrase:** Required if a passphrase was specified during key creation. Your Oracle Cloud Infrastructure Object Storage private key passphrase.

5. Click **Save**.

The values you entered must match your Oracle Cloud Infrastructure credentials. If you get an error message, verify your entries, update any incorrect values, and click **Save** again.

6. Click **Show Advanced File System Configuration**.
   Enter the required configuration information or click **Use Default** to accept the default values:
   - **NFS Allowed Hosts:** A comma-separated list of hosts allowed to connect to the NFS export. You can also specify ***** to allow all hosts to connect.
     For example: `2001:db8:9:e54::/64`, `192.0.2.0/24`
   - **NFS Export Options:** The NFS export options.
     Example: `rw`, `sync`, `insecure`, `no_subtree_check`, `no_root_squash`

     Important

     Do not specify the `fsid` option.
   - **Concurrent Uploads:** The number of concurrent uploads to Oracle Cloud Infrastructure.
     This field indicates the maximum number of files that can be concurrently uploaded in Storage Gateway. If the value is 15, the concurrent file uploads can be between 1-15.
     The allowed range is from 1 to 30.
   - **Sync Policy:** The metadata operations are flushed to the disk based on the sync policy, but do not affect on-disk consistency. Currently, only **Posix Standard** is supported. Only the synchronous transactions (like fsync, ODSYNC, and OSYNC) are committed to the disk. All other transactions are handled asynchronously.
   - **Cloud Read-ahead**: The number of blocks to be downloaded and used to *read ahead* when reading files for improved performance.
     Default value: 50

- **Maximum Read Cache Size in GiB:** The maximum read cache.
  When the read cache is full or reaches the configured limit, Storage Gateway removes files from the cache based on a least recently used (LRU) algorithm. Files pending upload to your tenancy are not removed from cache. You can also preserve files that you do not want removed from cache.

  Note

  The number of files in cache is limited to 20,000, regardless of the specified cache size in bytes.

  See Configuring the Cache for File Systems for details.
  The default value depends on how you provisioned local storage before installing Storage Gateway. The recommended local storage disk size is 600 GB (500 GB for file system cache, 80 GB for metadata, 20 GB for log). If you followed the documented recommendations, the default value for the read cache is approximately 300 GB.

7. Click **Save**.
   The file system is created and appears in the **File Systems** listing.

Connecting a File System

After you create a file system, you must connect the file system to an Oracle Cloud Infrastructure Object Storage bucket before you can store and retrieve data through the file system.

To connect a file system

1. Log in to the Storage Gateway management console.

2. On the **Dashboard** tab, identify the file system that you want to connect to your Object Storage bucket.

   Click **Connect**.
3. If a bucket with the same name as the file system exists in Object Storage, the file system is connected to that bucket. Any existing data cached in the Storage Gateway file system is deleted to ensure consistency with the data stored in the bucket. If a bucket by that name doesn't exist, the bucket is created and the file system is connected to the bucket. If the compartment OCID was specified during file system creation, then the bucket is created in that compartment. Otherwise, the bucket is created in the `root` compartment by default.



Important

You can mount a read/write file system on only one Storage Gateway at a time.

If the file system that you're importing is connected to another Storage Gateway, the **File System: Claim Ownership** window appears. You can claim ownership and confirm that the other Storage Gateway can be disconnected. Enter the following information, and then click **Claim Ownership**:

- Public key finger print
- Private key
- Private key passphrase

Claiming ownership ensures that you don't inadvertently connect a new file system to a bucket that's already connected to another Storage Gateway file system.

Mounting File Systems on Clients

Each Storage Gateway file system maps a directory to an Oracle Cloud Infrastructure Object Storage bucket. To establish the connection between Storage Gateway and an NFS client, you must mount the Storage Gateway file system on the NFS client.

Any Linux/UNIX NFS client certified to work with NFSv4 server running on Oracle Linux 7.x is compatible with Storage Gateway.



Note

Storage Gateway does not currently support NFS clients running on Windows or Mac OS.

To mount a Storage Gateway file system

1. Log in to the Storage Gateway host.
2. Start Storage Gateway:

   ```
   sudo ocisg up
   ```

3. Find the NFS port number:

   ```
   sudo ocisg info
   ```

   Note the NFS port number from the output. For example:

   ```
   Management Console: https://myStorageGatewayHost.example.com:32775

   If you have already configured a OCISG File System via the Management Console, you can access the
   NFS share using the following port.

   NFS Port: 32774

   Example: mount -t nfs -o vers=4,port=32774 myStorageGatewayHost.example.com:/<OCISG File System
   name> /local_mount_point
   ```

In the sample output:

- myStorageGatewayHost.example.com is the Storage Gateway host name.
- 32775 is the management console port number.
- 32774 is the NFS port number.

4. Log in to the NFS client from which you want to access your service instance through Storage Gateway.
5. Create a directory on the NFS client.
6. Mount the file system on the directory that you created on the NFS client:

```
sudo mount -t nfs -o vers=4,port=<NFS_port_number> <storage_gateway_host_name>:/<ocisg_file system_name> /<local_mount_point>
```

In this command:

- Replace *<NFS_port_number>* with the NFS port number you noted earlier.
- Replace *<storage_gateway_host_name>* with the server name or IP address of the server on which Storage Gateway is installed.
- Replace *<ocisg_file system_name>* with the file system name that you want to mount.
- Replace *<local_mount_point>* with the path to the directory you created on the NFS client.

For example:

```
sudo mount -t nfs -o vers=4,port=32774 myStorageGatewayHost.example.com:/myFirstFS /home/xyz/abc
```

In this example,

- `32774` is the NFS port.
- `myStorageGatewayHost.example.com` is the Storage Gateway host name.
- `myFirstFS` is the file system name.
- /home/xyz/abc is the path to the directory `abc` on the NFS client.

The Storage Gateway file system is now mounted on the NFS client directory. You can now access the Storage Gateway file system from the NFS client.

For more information, see Using Storage Gateway File Management Operations .

Viewing the Details of a File System

You can view the configuration details of a file system and monitor the upload activity through the management console of Storage Gateway.

To view the details of a file system

1. Log in to the management console.
2. Click the name of the file system.
    - The **Details** tab displays the Oracle Cloud Infrastructure service type, storage tier, and the identity domain associated with your tenancy. If the file system is connected, you can see mount point connection information to help you with mounting that file system. For example:
      NFS Client Mount Command: mount -t nfs -o vers=4,port=*<nfs_mount_port>* 129.213.122.84:/perftest01 /*<local_mount_point>*
    - The **Settings** tab displays the following details:
        - Details of the tenancy and scope specified for the file system.
        - File system properties.
        - NFS and cache settings for the file system.

      You can edit these settings. If you make changes, remember to click **Save**.

      If your file system is connected, you can also see:
    - The **Activity** tab, which shows ongoing and pending file upload activity.
      If you contact Oracle Support Services about any issue with the file system, you might need to provide the file system log to help diagnose the issue. To view or

download the file system log, click **View Streaming Logs** near the lower-right corner of the **Details** tab.

- The **Completed Uploads** tab, which shows the last 100 files that were uploaded to Oracle Cloud Infrastructure Object Storage during the current browser session.



Note

The file list doesn't persist across browser sessions. If you refresh the page or open the **Completed Uploads** tab in another browser after the files are uploaded, the list will be empty.

- You can also disconnect the file system. See Disconnecting a File System.

Changing the Properties of a File System

You can change the properties of a file system using the Storage Gateway management console.

To change the properties of a file system

1. Log in to the management console.
2. In the **Dashboard**, click the name of the file system that you want to edit.
3. In the **Settings** tab, edit the file system properties and advanced settings, such as the cache limits.
4. Click **Save**.
5. For the changes to take effect, disconnect and reconnect the file system.

Disconnecting a File System

When a file system is disconnected, no one can access or modify that file system.

We recommend disconnecting file systems that are not in use. Disconnecting a file system frees up the resources associated with that file system, making those resources available to file systems that are active (connected).

To disconnect a file system

1. Log in to the management console.
2. In the **Dashboard**, click the name of the file system that you want to disconnect.
3. Click **Disconnect**.
   When you disconnect a file system, the bucket to which the file system was previously connected and the contents of that bucket remain intact.
4. For the changes to take effect, disconnect and reconnect the file system.

You can resume storing and retrieving data by connecting the file system again. You can delete the disconnected file system when you no longer need it. For more information, see Deleting a File System.

Importing an Existing File System

Before you import an existing file system from another Storage Gateway, ensure that any pending file uploads to Oracle Cloud Infrastructure Object Storage are complete.

To import an existing file system

1. Log in to the management console.
2. Click the **Create File System** navigation link.
3. Click **Create File System** in the navigation pane on the left.
   The **Create a File System** page appears.

4. Enter the required information in **Create a File System**.
   For the file system name, enter the name of the existing file system that you want to import to this Storage Gateway.
5. Click **Save**.
6. Select the options that you'd like to enable in the file system.
7. Click **Show Advanced** and enter the required information.
8. Click **Save**.
   The file system is created and appears on the **Dashboard** tab.
9. Click **Connect** for the file system that you want to import.
   If the file system that you're importing is connected to another Storage Gateway, the **File System: Claim Ownership** window appears so you can claim ownership and confirm that the other Storage Gateway can be disconnected. Enter the following information and click **Claim Ownership**:
   - Public key finger print
   - Private key
   - Private key passphrase
10. Mount the file system to a directory on the Storage Gateway host and set up the NFS export. For example:

    ```
    sudo mount -t nfs -o vers=4,port=<NFS_port_number> <storage_gateway_host_name>:/<ocisg_file system_name> /<local_mount_point>
    ```

    In this command:
    - Replace *<NFS_port_number>* with the NFS port number you noted earlier.
    - Replace *<storage_gateway_host_name>* with the server name or IP address of the server on which Storage Gateway is installed.
    - Replace *<ocisg_file system_name>* with the file system name that you want to mount.
    - Replace *<local_mount_point>* with the path to the directory you created on the NFS client.

Deleting a File System

You can delete a file system from Storage Gateway when you no longer need it.

To delete a file system:

1. Log in to the management console.
2. On the **Dashboard**, identify the file system that you want to delete.



Important

When you disconnect a file system, the bucket to which the file system was previously connected and the contents of that bucket remain intact.

Deleting a file system does not automatically delete the objects in the bucket. If you want to remove objects from the Object Storage bucket, set the **Delete Old File Versions** property for the file system and delete all the files before disconnecting the file system.

3. Ensure that the file system is disconnected. If it's still connected, click **Disconnect**.
4. After the file system is disconnected, click its name.
   The details of the file system appear.
5. Click **Delete**.
   The file system is deleted from Storage Gateway.

Managing Storage Gateway

This topic describes some basic Storage Gateway management tasks.

Managing Storage Gateway Using the CLI

You can use the `ocisg` command line interface (CLI) to manage Storage Gateway. To use the CLI, open an `ssh` connection and log in to the host on which you installed Storage Gateway.



Note

You can use the `ocisg` command line interface (CLI) to create, manage, and monitor Storage Gateway Cloud Sync jobs. See Using Storage Gateway Cloud Sync for details.

The CLI supports the following Storage Gateway management tasks:

- To start Storage Gateway:

```
sudo ocisg up
```

- To stop Storage Gateway:

```
sudo ocisg down
```



Note

If the server with a Storage Gateway system fails, you can reinstall and start a new one. All the configuration and system data is automatically downloaded and applied. The pending upload and download activities resume when the new Storage Gateway system runs.

If a disk cache is unrecoverable on the Storage Gateway server, data might be lost since the file might not have been transferred to the bucket in your tenancy. To ensure efficient data protection, see Best Practices for Using Storage Gateway.

- To view details about Storage Gateway and how to access the management console:

```
sudo ocisg info
```

- To find the version of Storage Gateway:

```
sudo ocisg version
```

- To configure Storage Gateway to use a proxy server for connections to Oracle Cloud Infrastructure Object Storage:

```
sudo ocisg configure proxy <proxy_server_URL>
```



Note

After configuring the proxy server, you must stop and restart Storage Gateway.

By default, no proxy server is specified.

- To remove previously configured proxy server details in Storage Gateway:

```
sudo ocisg configure proxy [remove]
```

- To configure Storage Gateway to use SSL when communicating with the management console and REST APIs:

  ```
  sudo ocisg configure ssl true
  ```

  SSL is enabled by default.

  Note

  After configuring Storage Gateway to use SSL, you must stop and restart Storage Gateway.

  To disable SSL:

  ```
  sudo ocisg configure ssl false
  ```

- To specify ports for the Storage Gateway services:

  ```
  sudo ocisg configure port <service> <port_number>
  ```

  - *<service>*: Specify `admin`, `nfs`, or `rest`.
  - *<port_number>*: Ensure that the port number you specify is not already in use on the Storage Gateway host.

  By default, the port number is assigned dynamically for the Storage Gateway services when you start Storage Gateway.

  Note

  For the port assignment to take effect, you must stop and start Storage Gateway.

- To remove the static port assignment for a service:

  ```
  sudo ocisg configure port <service> remove
  ```

- To allocate memory for the Storage Gateway host:

```
sudo ocisg configure memory <memory_in_GB>
```

- To remove the memory allocation:

```
sudo ocisg configure memory remove
```

By default, Storage Gateway uses 4 GB of the available memory on the host server. You can delete the memory information by using the `remove` parameter.



Note

After configuring memory for Storage Gateway, you must stop and restart Storage Gateway.

- To specify the docker network mode:

```
sudo ocisg configure network mode
```

The mode can be either *host* or *bridge*.

The default mode is `bridge`. In this mode, you can run multiple instances of Storage Gateway on your host.

In the `host` mode, you can run only a single instance of Storage Gateway. Network performance is better in `host` mode.



Note

After specifying the docker network mode, you must stop and restart Storage Gateway.

- To change the Storage Gateway `admin` password:

```
sudo ocisg do password:reset
```

Set a new password:

```
sudo ocisg password:set <new_password>
```

Enter a password that meets the following requirements:

- Uses from 8 to 32 characters.
- Includes at least one special character, one numerical character, one uppercase character, and one lowercase character.

- To view help for the available commands:

```
sudo ocisg help
```

Using Storage Gateway File Management Operations

This topic describes how to use the Storage Gateway file management operations.

Exercise caution when using the REST API, Java library, or any other client to retrieve, create, update, or delete objects directly in a bucket that's mapped to a file system in Storage Gateway. Until you **Refresh** the Storage Gateway file system, Storage Gateway is not aware of the changes and data is inconsistent between Storage Gateway and Object Storage.

Uploading Files to Buckets

Before you connect the file system to the Oracle Cloud Infrastructure Object Storage bucket, make a note of the Oracle Cloud Infrastructure Object Storage tenancy details such as namespace, tenant OCID, and compartment OCID.

Copy the files to the mounted directory on the Storage Gateway or the NFS client host. Storage Gateway writes the files to the disk cache. The system queues and asynchronously uploads the files to an Object Storage bucket. Corresponding objects are created in the storage tier you specified during file system creation, either **Standard** or **Archive**. See Creating Your First File System or Managing File Systems for details.



Note

Storage Gateway automatically performs multipart upload for files larger than 128 MB. See Using Multipart Uploads for details.

You can view files uploaded to your tenancy during the current browser session. See the **Completed Uploads** tab in Viewing the Details of a File System.

Reading Files

When you write a file to a Storage Gateway file system, the system stores the file in the local disk cache. You can read the file directly from the mounted directory. Storage Gateway asynchronously copies the file to the corresponding Object Storage bucket in your tenancy. To retrieve the data from the bucket using Storage Gateway, read the files from the mounted directory.

If space is available, Storage Gateway automatically places the files in the read cache. If the file is in the read cache, you can retrieve the file immediately. If the file is not available in the read cache and it is stored in the `Archive` tier, you must restore the object. For more information, see Restoring Files and Objects from Archive Storage.



Note

You cannot read or write to a file that is stored in the **Archive** tier and does not exist in the read cache. This action returns an `Input/Ouput error`.

Storage Gateway checks data integrity using checksum verification on uploads. The system might not be able to perform data integrity validation on a partial read, since checksum verification works only on a whole file or object.

To read the upload checksum for a file in a file system, run the following command from the NFS client on which the file system is mounted:

```
sudo docker exec ocisg bash -c "cd /mnt/gateway/${filesystem} && cat ${filepath}:::meta:csm"
```

Restoring Files and Objects from Archive Storage

You can initiate a file restore from the Storage Gateway command line. You can also initiate an object restore from Archive Storage in Oracle Cloud Infrastructure. You can read the corresponding file using Storage Gateway after the object has been restored to Object Storage.



Note

Storage Gateway supports Oracle Cloud Infrastructure Object Storage object lifecycle policies to manage the archiving and deletion of objects in a bucket according to a pre-defined schedule. Using object lifecycle policies, you can specify bucket creation in the Standard Object Storage tier, and then create a policy to schedule the subsequent movement of data to the Archive Storage tier. This lifecycle policy archival method is useful if you have on-premises applications that generate intermediary or temporary files and directories that are inappropriate for immediate archival. See Using Object Lifecycle Management for details.

Restoring Archived Files Using the Storage Gateway Command Line

To restore one or more archived files

Open a command prompt on the Storage Gateway host and run the `ocisg archive restore`

command. Specify the full path to a directory or to a file.

```
ocisg archive restore <file_system_name> <full/path/to/directory/or/file> [<#_of_hours>]
```

By default, the file remains restored for 24 hours after restoration. However, you can optionally specify [*<#_of_hours>*] with an integer value of from 1 to 240 hours.

For example, to restore all the files in a directory:

```
ocisg archive restore myFS myDir/mySubDir 240
```

For example, to restore a single file:

```
ocisg archive restore myFS myDir/mySubDir/file2 240
```

To check the archive status of one or more files

You can get the archive status for all files in a file system, for all files in a directory, or for an individual file.

Open a command prompt on the Storage Gateway host and run the `ocisg archive restore-status` command.

```
ocisg archive restore-status <file_system_name> [<full/path/to/directory/or/file]
```

The status can be one of the following:

- Archived
- In progress
- Restored

For example, to check the archive status for all files in a file system:

```
ocisg archive restore-status myFS
```

To check the archive status for all files in directory:

```
ocisg archive restore-status myFS myDir/mySubDir
```

To check the archive status for an individual file:

```
ocisg archive restore-status myFS myDir/mySubDir/file2
```

To check the restoration job status for a file system

You can get the status for all restoration jobs that have been initiated for a file system.

Open a command prompt on the Storage Gateway host and run the `ocisg archive restore-jobs` command.

```
ocisg archive restore-jobs <file_system_name>
```

For example:

```
ocisg archive restore-jobs myFS
```

Restoring Archived Files Using Oracle Cloud Infrastructure



Important

If you use Oracle Cloud Infrastructure to restore archived objects, use the **Refresh** operation in Storage Gateway to display the data that was added or modified directly in Object Storage.

To restore an archived object using the Oracle Cloud Infrastructure Console



Tip

You need OBJECT_RESTORE permissions to restore Archive Storage objects.

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
2. Choose the compartment your bucket is in.
   A list of buckets is displayed.
3. Click the bucket name that contains your object.

4. Click **Objects** under **Resources**.
   A list of objects in the bucket is displayed.
5. To restore a single object, click the Actions icon (three dots) to the right of the object you want to restore, and then click **Restore**. To restore multiple objects, select the check boxes to the left of each object you want to restore, then click **Restore**.
6. Optionally, specify the **Time Available for Download in Hours**.
   By default, you have 24 hours to download an object after restoration. However you can alternatively specify a download time of from 1 to 240 hours. You can find out how much download time is remaining by looking at **Available for Download** in object **Details** or by looking at the Actions icon (three dots) menu to the right of **Download**. Refresh the browser to obtain up-to-date remaining download time information.
   After the allotted download time expires, the object returns to Archive Storage.
7. Click **Restore Objects**.
   Error messages are generated if there is a problem with restoring the selected objects. You can optionally click **Retry failed restore option.**

To check the status of an object restoration using the Oracle Cloud Infrastructure Console

1. Open the navigation menu. Under **Core Infrastructure**, click **Object Storage**.
2. Choose the compartment your bucket is in.
   A list of buckets is displayed.
3. Click the bucket name that contains your object.
4. Click **Objects** under **Resources**.
   A list of objects in the bucket is displayed.
5. Click the Actions icon (three dots) to the right of the object you want to check the restoration or download status of, then click **Details**.
6. Check the **Status**.
   Status displays one of the following:

- Archived
- Restoring
- Restored

To restore an archived object using the Oracle Cloud Infrastructure CLI



Tip

You need OBJECT_RESTORE permissions to restore Archive Storage objects.

Open a command prompt and run `oci os object restore` to restore an object from Archive Storage:

```
oci os object restore -ns <object_storage_namespace> -bn <archive_bucket_name> --name <archived_object_name> [--hours <#_of_hours>]
```

By default, you have 24 hours to download an object after restoration. However, you can optionally specify `--hours` with an integer value of download time of from 1 to 240 hours.

To check the status of an object restoration using the Oracle Cloud Infrastructure CLI

Open a command prompt and run `oci os object restore-status` to check the status of restoring an object from Archive Storage:

```
oci os object restore-status -ns <object_storage_namespace> -bn <archive_bucket_name> --name <archived_object_name>
```

Deleting Files

Remove the files that you no longer need from the NFS client by deleting them from the directory on which the file system is mounted.

Monitoring Storage Gateway

This topic describes how to monitor Storage Gateway file system upload activity, system health, and storage usage. The topic also describes how to view system notifications and how to receive notifications in email.



Important

Monitoring file system activity in the Storage Gateway management console consumes system resources. Monitoring file system activity is not recommended when Storage Gateway is under high load.

Monitoring Upload Activity

When you upload a file to a file system, you can view the status of the upload activity. The **Activity** tab shows the ongoing and pending upload activity in a file system.

To monitor upload activity

1. Log in to the management console.
2. Click **File Systems** in the upper-right corner of the management console.
3. On the **File Systems** page, select a file system.
4. Click **Activity**.
   You can see the upload progress of the file in the **Uploading** pane.

Monitoring System Health Status

You can monitor the overall system health status using **System Status** on the right side of the management console.

Storage Gateway performs an automated "health check" on the system to monitor the status of:

- Storage Gateway resources and services.
- Local storage, file system cache, metadata storage, and log storage.

System Status shows the result of Storage Gateway health analysis, either **Healthy** or **Unhealthy**.

System Status also provides information on:

- Any system warnings or errors.
- **Local I/O Mode**, which depends on local disk usage:
    - **Normal**
      The available disk space is greater than 10 GB in Storage Gateway. You can upload files in Storage Gateway and upload them to your Oracle Cloud Infrastructure tenancy.
    - **Rejecting I/O**
      The available disk space is less than 10 GB in Storage Gateway. Storage Gateway runs in protection mode and does not allow any writes to its local disk. All read operations work normally. All Storage Gateway metadata operations fail except for deletions and truncation.
      To return to **Normal** mode, you must wait until all ongoing upload activities complete and the files are removed from the read cache.
- **Throughput**
  The approximate upload throughput to Object Storage. If there is no recent activity, **Throughput** shows **Idle**.
- **Available Read Cache**
  The amount of read cache available. For optimal performance, reliability, and fault tolerance, follow the guidelines for configuring cache storage . See Configuring Local Storage for File Systems and Cache for details.
- **Pending Uploads**

The number of files or directories, for all file systems, awaiting upload to Object Storage. If **Pending Uploads** is 0 (zero), all files and directories have been uploaded.

Monitoring Storage Usage

You can track the storage usage and availability.

To monitor storage usage

1. Log in to the management console.
2. Click **System** in the upper-right corner of the management console.
3. Click **System Stats**.
   The system data appears in three panes:
   - **Local Storage**
   - **Local I/O**
   - **Local Resources**

Local Storage

This pane provides a graphical representation of the amount of storage being used and the available free storage on the Storage Gateway host. You can see:

- Available local storage.
- Storage used for pending uploads and preserved cache files.
- Storage used for metadata.
- Storage used for logging.
- Storage used for other applications.

Local I/O

This pane displays the local I/O mode of Storage Gateway based on the local disk space usage on the Storage Gateway host.

Local Resources

This pane shows the overall memory usage and availability for Storage Gateway from the following fields:

- **Available Cores:** The number of CPUs being used by Storage Gateway.
- **Maximum Memory Available to Storage Gateway:** The total RAM available for Storage Gateway.
- **Memory Used by Storage Gateway:** The amount of memory being used by the file systems in Storage Gateway.
- **Free Memory:** The amount of free RAM available in Storage Gateway host.

Viewing System Notifications

The **System Notifications** tab shows system notifications and helps you track overall system performance.

To view system notifications

1. Log in to the management console.
2. Click **System** in the upper-right corner of the management console.
3. Click **System Notifications**.
   You can view a list of warnings or critical system notifications.

Configuring Email Notification

You can configure Storage Gateway to notify you about system health by email.

To configure email notification

1. Log in to the management console.
2. Click **System** in the upper-right corner of the management console.
3. Click **System Notifications**.
4. If email notifications are not yet configured, click **Click here to configure**.
5. Enter the required information for the following fields:
   - SMTP server.
   - Email addresses to receive notifications.
6. Click **Show Advanced Options** and enter the required information in the advanced configuration fields:
   - SMTP port.
   - SMTP User name.
   - SMTP Password.
   - Sender's Email Address.
     The default value is: `noreply@oracle.com`
7. Click **Save**.
8. Click **Test Email Notification** to verify that the specified email address receives a system notification email.

Using Storage Gateway Cloud Sync

Use Cloud Sync to move on-premises datasets from a local file system to Storage Gateway, where the data is then moved asynchronously to Oracle Cloud Infrastructure Object Storage. You can also use Cloud Sync to synchronize Storage Gateway file system changes back to the local file system.

Cloud Sync generates the following for each sync job:

- Sync status (Created, Running, Completed, Failed, or Canceled).
- Number of files and directories to be copied from the source to the target.
- File and directory upload progress .
- Number of files and directories uploaded to the target.
- Time the job started, the time the job ended, and the run duration.

You can use the Storage Gateway management console or the command line interface (CLI) to create, manage, and monitor Cloud Sync jobs.

About Cloud Sync

Cloud Sync is an enhanced wrapper around the Linux `rsync` command and relies on `rsync` to detect new and changed files using size and modification time. Cloud Sync also relies on `rsync` to verify the files once the data transfer is complete using checksums of the files. Cloud Sync calls the Storage Gateway diagnostic commands to provide detailed data movement progress and status between your local server, Storage Gateway, and Oracle Cloud Infrastructure.

Understanding Failure Behavior

Because the file system is mounted locally, the NFS client running on the host handles any issues resulting from file system availability or connectivity.

Cloud Sync does the following:

- Reports and logs any failures to list or copy specific files (for example, resulting from permission issues).
- Monitors and reports on the pending and completed uploads to Object Storage.

Storage Gateway handles any connectivity and access issues to Object Storage and performs retry operations as needed.

Prerequisites for Cloud Sync

- Create the Storage Gateway file system that serves as either the source or target destination for the sync operation. A file system on the Storage Gateway host maps to a bucket with an identical name in Oracle Cloud Infrastructure Object Storage. See Creating Your First File System or Adding a File System for details.
- Obtain the proper credentials to mount the file system share from the local server.
- The local server source must:
    - Have names services set up correctly so that UIDs and GIDs are preserved and are not remapped to `nobody`.
    - Be exported with root permissions to read and traverse the entire source tree.

Using the Management Console

You can use the Storage Gateway management console to create, manage, and monitor Cloud Sync jobs.

To create a Cloud Sync job that syncs all files and directories at the specified source location

1. Log in to the management console.
2. Click **Cloud Sync** in the upper-right corner of the management console.
   By default, a list of all the Cloud Sync jobs that have already been created is displayed.
3. Click **Create Cloud Sync Job**.
4. In **Create Cloud Sync Job** page, specify the attributes of the job:
    - **Job Name**: Required. A unique, user-friendly name for the job. Avoid entering confidential information.
    - **Source Path**: Path to the Cloud Sync source directory containing the files to sync.

- If your are syncing a local file system to Oracle Cloud Infrastructure, specify the source path as:

  ```
  /cloudsync/mounts/<user_mount>[/<path_to_directory>]
  ```

- If your are syncing Oracle Cloud Infrastructure to a local file system, specify the source path as:

  ```
  <storage_gateway_file_system>/<path_to_directory>]
  ```

- **Target Path**: Path to the Cloud Sync target directory for the synced files.
  - If your are syncing a local file system to Oracle Cloud Infrastructure, specify the target path as:

    ```
    <storage_gateway_file_system>/<path_to_directory>]
    ```

  - If your are syncing Oracle Cloud Infrastructure to a local file system, specify the target path as:

    ```
    /cloudsync/mounts/<user_mount>[/<path_to_directory>]
    ```

- **Enable Auto-Deletion:** Enable this option if you want Cloud Sync to automatically delete files from the target when:
  - Files are deleted from the source.
  - Source files have been renamed.

  By default, when a file is deleted on the source, Cloud Sync does not automatically delete the file on the target unless you enable **Enable Auto-Deletion**. Also, when a source file is renamed, the file with the old name is deleted and a file with the new name is created. By default, Cloud Sync does not delete the file with the old name on the target (retaining both a file with the old name and a file with the new name) unless you choose **Enable Auto-Deletion**. The names of all deleted files are stored in a job-specific log directory.

5. Click **Create Cloud Sync Job**.

A Cloud Sync job is created and displayed in the list of jobs.

To list Cloud Sync jobs

1. Log in to the management console.
2. Click **Cloud Sync** in the upper-right corner of the management console.
   By default, a list of all the Cloud Sync jobs is displayed.
3. Optionally, you can filter the job listing by status (Created, Running, Completed, Failed, or Canceled) and type of Cloud Sync job (upload or download) by clicking one of the **Quick Filters**.

To run a Cloud Sync job

1. Log in to the management console.
2. Click **Cloud Sync** in the upper-right corner of the management console.
3. In the list of jobs, find the Cloud Sync job that you want to run.
4. Click **Run** to the right of the job name.
   Cloud Sync runs the job. The management console displays the status of the job just below the job name.

To get the status of a Cloud Sync job

1. Log in to the management console.
2. Click **Cloud Sync** in the upper-right corner of the management console.
3. In the list of jobs, find the Cloud Sync job for which you want status.
   The management console displays the status of the job (Created, Running, Completed, Failed, or Canceled) just below the job name.

To cancel a Cloud Sync job

You can only cancel a job if it is running.

1. Log in to the management console.
2. Click **Cloud Sync** in the upper-right corner of the management console.

3. In the list of jobs, find the Cloud Sync job that you want to cancel.



Tip

Use **Quick Filters** to get a list of **Running** jobs.

4. Click **Cancel** to the right of the job name.

To delete a Cloud Sync job

You cannot delete a running job.



Tip

Cancel a running job before you try to delete it.

1. Log in to the management console.
2. Click **Cloud Sync** in the upper-right corner of the management console.
3. In the list of jobs, find the created, canceled, or failed Cloud Sync job that you want to delete.
4. Click **Delete** to the right of the job name.

Using the CLI

You can use the `ocisg` command line interface (CLI) to create, manage, and monitor Cloud Sync jobs. Using `ssh`, log in to the host on which you installed Storage Gateway to use the CLI.

To create a Cloud Sync job that syncs all files and directories at the specified

source location



Warning

Avoid entering confidential information in the job name.

Open a command prompt and run `ocisg cloudsync create` to create a job:

```
sudo ocisg cloudsync create [--auto-delete] [--parallel=<number>] [--files-from=<file>] <job_name>
<source_path> <target_path>
```

`--auto-delete` is an option to direct Cloud Sync to automatically delete files from the target when files are deleted from the source, and old names of files that have been renamed. By default, Cloud Sync does not automatically delete the files on the target unless you specify this option. The names of all deleted files are stored in a job-specific log directory.

--parallel=*<number>* is an option to specify the number of processes for data synchronization. By default, the number of processes is set to one. With a single process using a hard disk drive, you can expect a sync rate of 60-70 MB/s. If your system has higher disk throughput, you can use the number of processes that is proportional to the available bandwidth. Oracle recommends from two to five processes for optimal performance. The maximum number of processes allowed is 10.

--files-from=*<file>* is an option to specify a set of files that you want to sync to the target. If you do not set this option, the service syncs all files. The *<file>* should be a file under the `/clousdync/` directory. For example, `--files-from="/cloudsync/files.list"`.

The Cloud Sync job is created and displayed in the list of jobs.

To list Cloud Sync jobs

Open a command prompt and run `ocisg cloudsync list` to list jobs:

```
sudo ocisg cloudsync list [-s <status>] [<job_name_or_type>]
```

Optionally, you can filter the list of jobs by specifying job status (Created, Running, Completed, Failed, or Canceled). You can also list a single job by specifying the name of that job.

For example:

```
sudo ocisg cloudsync list sync_to_os1
```

To run a Cloud Sync job

Open a command prompt and run `ocisg cloudsync run` to run a job:

```
sudo ocisg cloudsync run <job_name>
```

For example:

```
sudo ocisg cloudsync run sync_to_os1
```

To get the status of a Cloud Sync job

Open a command prompt and run `ocisg cloudsync status` to get the status of a job:

```
sudo ocisg cloudsync status <job_name>
```

For example:

```
sudo ocisg cloudsync status sync_to_os1
```

To cancel a Cloud Sync job

You can cancel a job only if the job is in progress.

Open a command prompt and run `ocisg cloudsync cancel` to cancel a job:

```
sudo ocisg cloudsync cancel <job_name>
```

For example:

```
sudo ocisg cloudsync cancel sync_to_os1
```

To delete a Cloud Sync job

Open a command prompt and run `ocisg cloudsync delete` to delete a job:

```
sudo ocisg cloudsync delete <job_name>
```

For example:

```
sudo ocisg cloudsync delete sync_to_os1
```

Best Practices for Using Storage Gateway

Apply the recommendations found in the following topics to optimize the manageability, performance, reliability, and security of your Storage Gateway.

- Security Considerations
- Understanding Storage Gateway Performance
- Configuring Local Storage for File Systems and Cache
- Determining File System Cache Size
- Recommended Uses and Workloads
- Uses and Workloads Not Supported
- Renaming Directory Trees
- Limits on Storage Gateway Resources

Troubleshooting Storage Gateway

This topic covers some common Storage Gateway issues and how to address them.

I installed docker and NFS on my host, but I can't install Storage Gateway

1. Add the docker group to the existing groups in your host:

   ```
   sudo groupadd docker
   ```

2. Add your user id to the docker group:

   ```
   usermod -a -G docker <username>
   ```

3. Shut down your host:

```
shutdown -r now
```

4. Log in to your host and run the Storage Gateway installation script:

```
sudo ./ocisg-install.sh
```

I can't access the management console

1. Run the `ocisg info` command:

```
sudo ocisg info
```

If Storage Gateway is not running, start Storage Gateway:

```
sudo ocisg up
```

2. Make a note of the management console port number from the output:

```
Management Console: https://myStorageGatewayHost.example.com:32775

If you have already configured a OCISG File System via the Management Console, you can access the
NFS share using the following port.

NFS Port: 32774

Example: mount -t nfs -o vers=4,port=32774 myStorageGatewayHost.example.com:/<OCISG File System
name> /local_mount_point
```

In the example, `myStorageGatewayHost.example.com` is the Storage Gateway host name and `32775` is the management console port number.

3. Ensure that Storage Gateway is running `docker` on the Storage Gateway host.
4. Check that the management console port number in the output from `ocisg info` matches the port you're using to access the management console.
5. Ensure that you are using `https` if you have enabled SSL. SSL is enabled by default.

I am unable to mount a file system

1. Run the `ocisg info` command:

   ```
   sudo ocisg info
   ```

   If Storage Gateway is not running, start Storage Gateway:

   ```
   sudo ocisg up
   ```

2. Make a note of the management console port number and NFS port number from the output:

   ```
   Management Console: https://myStorageGatewayHost.example.com:32775

   If you have already configured a OCISG File System via the Management Console, you can access the
   NFS share using the following port.

   NFS Port: 32774

   Example: mount -t nfs -o vers=4,port=32774 myStorageGatewayHost.example.com:/<OCISG File System
   name> /local_mount_point
   ```

   In the sample output:

   - myStorageGatewayHost.example.com is the Storage Gateway host name.
   - 32775 is the management console port number.
   - 32774 is the NFS port number.

4. Log in to the NFS client from which you want to access your service instance through Storage Gateway.
5. Create a directory on the NFS client.
6. Mount the file system on the directory that you created on the NFS client:

   ```
   sudo mount -t nfs -o vers=4,port=<NFS_port_number> <storage_gateway_host_name>:/<ocisg_file
   system_name>/<local_mount_point>
   ```

   In this command:

- Replace *<NFS_port_number>* with the NFS port number.
- Replace *<storage_gateway_host_name>* with the server name or IP address of the server on which Storage Gateway is installed.
- Replace *<ocisg_file system_name>* with the name of the file system you want to mount.
- Replace *<local_mount_point>* with the path to the directory you created on the NFS client.

For example:

```
sudo mount -t nfs -o vers=4,port=32774 myStorageGatewayHost.example.com:/myFirstFS /home/xyz/abc
```

In this example,

- `32774` is the NFS port number.
- `myStorageGatewayHost.example.com` is the Storage Gateway host name.
- `myFirstFS` is the file system name.
- `/home/xyz/abc` is the path to the directory `abc` on the NFS client.

7. Ensure that Storage Gateway is running `docker` on the Storage Gateway host.
8. Ensure that the NFS protocol is running:

```
sudo systemctl enable nfs-server
```

9. Check that the NFS port number in the output from `ocisg info` matches the port you're using to connect to with your NFS client.

Additional NFS Troubleshooting

The Storage Gateway installation software installs the NFS, if needed, and automatically configures it. After the installation, the NFS is configured and a file system created. You can then mount the filesystem from a remote client. Sometimes this mount can fail.

To troubleshoot a mount failure:

1. Ensure that the NFS port is included in the Storage Gateway's subnet security list and that it is available there. If the port does not appear in the subnet security list, add it and retry the mount.
2. Run `rpcinfo -p`. The command should return:

   ```
   100003 4 tcp 2049 nfs
   ```

   This result means NFS is ready, available, and the mount will succeed.
3. If `nfs` does not appear in the response to the `rpcinfo -p` command, enable and restart both `rpcbind` and NFS:

   ```
   sudo systemctl enable rpcbind
   sudo systemctl enable nfs
   sudo systemctl start rpcbind
   sudo systemctl start nfs
   ```
4. Run the `rpcinfo -p` command again to verify that NFS is now available.
   a. If NFS still is not available, reboot the Storage Gateway.
   b. Run the `rpcinfo -p` command again to confirm.
5. If you remain unable to mount the file system, contact My Oracle Support.

Contacting Oracle Support

If you need technical support or help with Storage Gateway, you can go to My Oracle Support and create a service request. See Creating a Service Request for information.

Upgrading Storage Gateway

Storage Gateway notifies you when there is a new version available for you to download and install:

- A pop-up notification appears in the management console after you log in.
- A small banner notification appears at the top of the **Dashboard**.

- If you have configured email notifications, the system sends an email notification. See Configuring Email Notification for details.



Important

If you are upgrading from Storage Gateway 1.0, underlying database and schema changes require you to recreate your Storage Gateway file systems. See Recreating Your File Systems for details.

Before You Begin

- Plan for downtime appropriately since the upgrade takes some time to complete. The downtime varies depending on the system resources, the number of file systems, and the number of files.
- Wait for any pending or ongoing write operations from the NFS client instances to complete, then unmount all file systems.
- Wait for pending uploads to Oracle Cloud Infrastructure Object Storage to complete. On the **Dashboard** under **System Status**, ensure that the **Pending Uploads** field shows 0.
- Disconnect all of the file systems.
- Ensure that there is no ongoing activity in the **Activity** tab for each file system in the management console.

Upgrading Storage Gateway

To upgrade to Storage Gateway 1.2.1

1. Log in to your Storage Gateway host.

2. Download the Storage Gateway 1.2.1 tar archive.
3. Extract the files from the downloaded `ocisg-1.2.1.tar.gz` file:

   ```
   tar xvzf ocisg-1.2.1.tar.gz
   ```

   This command extracts the file's contents to a subdirectory named `ocisg-1.2.1`.
4. Change directory to `ocisg-1.2.1`:

   ```
   cd ocisg-1.2.1
   ```
5. Run the installation script as `sudo` or `root` user:

   ```
   sudo ./ocisg-install.sh
   ```

If you encounter any interruption during the upgrade, such as lost connectivity, rerun the installation script to resume the upgrade.

If you are upgrading from Storage Gateway 1.0, you must recreate the file systems that were created in the 1.0 version of the Storage Gateway software. Connect the file systems in the management console and claim ownership if there's a bucket ownership prompt. Storage Gateway version 1.2.1 rebuilds the local metadata for existing buckets in Object Storage. The more objects there are in the buckets, the more time it takes to rebuild the metadata.

Recreating Your File Systems

When you created file systems in Storage Gateway 1.0, corresponding Oracle Cloud Infrastructure Object Storage buckets were created. Recreate those file systems in Storage Gateway so that you can connect to the same buckets and automatically see the files that have already been uploaded to Object Storage.

When you recreate Storage Gateway file systems, data that you already uploaded to Oracle Cloud Infrastructure Object Storage is automatically included in the newly created file system.

To recreate your file systems

1. Log in to the management console.
2. Click **File Systems** on the upper-right area of the management console.
3. Click **Create File System**.
4. Enter the required information in **Create a File System**:
    - **File System Name:** A unique, friendly name for the file system. Avoid entering confidential information. Use the following guidelines when naming a file system:
        - Use from 1 to 256 characters.
        - Valid characters are letters (upper or lower case), numbers, hyphens, underscores, and periods. (**Important**: The name cannot contain a slash (/) character because this character delimits bucket and object names in Oracle Cloud Infrastructure Object Storage.)

          Note

          If an Object Storage bucket by this file system name doesn't exist in your tenancy, the bucket is created.

          If a corresponding Object Storage bucket by this file system name exists in your tenancy and there is data in the bucket, Storage Gateway works asynchronously to sync the bucket and file system contents.
    - **Select the Object Storage tier in which you want to store your data**



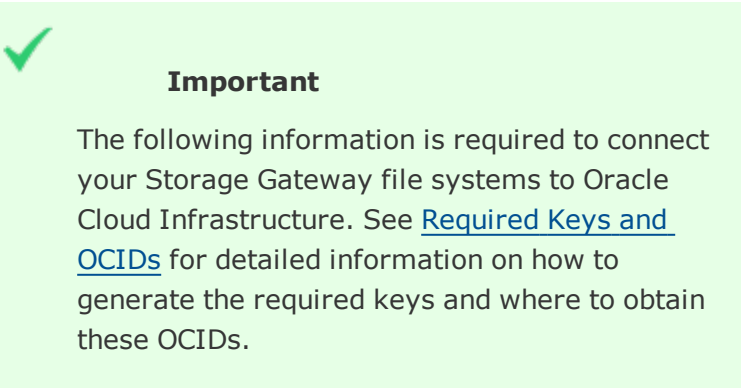
Important

Once set, you cannot change the storage tier in which a bucket resides.

You can use the Oracle Cloud Infrastructure Object Storage object lifecycle policies feature to manage the archiving and deletion of objects in a bucket according to a predefined schedule. See Using Object Lifecycle Management for details.

- **Standard:** The Standard tier is the primary default Object Storage tier for storing data that requires frequent and fast access. See Overview of Object Storage for more information.
- **Archive:** The Archive tier is a special tier for storing data that is accessed infrequently and requires long retention periods. See Overview of Archive Storage for more information. Access to data in the Archive tier is not immediate since you must restore archived data before it's accessible (see Restoring Files and Objects from Archive Storage).

- **Object Storage endpoint:** Required. The Object Storage API endpoint for your service instance. To find the object storage API endpoint for your Oracle Cloud Infrastructure Object Storage tenancy, see the API documentation for Oracle Cloud Infrastructure Object Storage.



- **Compartment OCID:** Required. Unique identifier of your Oracle Cloud Infrastructure Object Storage compartment.
- **Tenant OCID:** Required. Unique identifier of your Oracle Cloud Infrastructure Object Storage tenancy.
- **User OCID:** Required. Unique identifier of your Oracle Cloud Infrastructure Object Storage user.
- **Public Key's Finger Print:** Required. Your Oracle Cloud Infrastructure Object Storage public key fingerprint.
- **Private Key:** Required. Your Oracle Cloud Infrastructure Object Storage private key.
- **Private Key Passphrase:** Required if a passphrase was specified during key creation. Your Oracle Cloud Infrastructure Object Storage private key passphrase.

5. Click **Save**.
   The values you entered must match your Oracle Cloud Infrastructure credentials. If you get an error message, verify your entries, update any incorrect values, and click **Save** again.

6. Click **Show Advanced File System Configuration**.
   Enter the required configuration information or click **Use Default** to accept the default values:
   - **NFS Allowed Hosts:** A comma-separated list of hosts allowed to connect to the NFS export. You can also specify ***** to allow all hosts to connect.
     For example: `2001:db8:9:e54::/64`, `192.0.2.0/24`
   - **NFS Export Options:** The NFS export options.
     Example: `rw`, `sync`, `insecure`, `no_subtree_check`, `no_root_squash`

     Important

     Do not specify the `fsid` option.
   - **Concurrent Uploads:** The number of concurrent uploads to Oracle Cloud Infrastructure.
     This field indicates the maximum number of files that can be concurrently uploaded in Storage Gateway. If the value is 15, the concurrent file uploads can be between 1-15.
     The allowed range is from 1 to 30.
   - **Sync Policy:** The metadata operations are flushed to the disk based on the sync policy, but do not affect on-disk consistency. Currently, only **Posix Standard** is supported. Only the synchronous transactions (like fsync, ODSYNC, and OSYNC) are committed to the disk. All other transactions are handled asynchronously.
   - **Cloud Read-ahead**: The number of blocks to be downloaded and used to *read ahead* when reading files for improved performance.
     Default value: 50
   - **Maximum Read Cache Size in GiB:** The maximum read cache.
     When the read cache is full or reaches the configured limit, Storage Gateway removes files from the cache based on a least recently used (LRU) algorithm.

Files pending upload to your tenancy are not removed from cache. You can also preserve files that you do not want removed from cache.



Note

The number of files in cache is limited to 20,000, regardless of the specified cache size in bytes.

See Configuring the Cache for File Systems for details.

The default value depends on how you provisioned local storage before installing Storage Gateway. The recommended local storage disk size is 600 GB (500 GB for file system cache, 80 GB for metadata, 20 GB for log). If you followed the documented recommendations, the default value for the read cache is approximately 300 GB.

7. Click **Save**.
   The file system is created and appears in the **File Systems** listing.

Uninstalling Storage Gateway

This topic describes how to uninstall Storage Gateway.

Uninstalling

To uninstall Storage Gateway

1. Log in to the on-premises host or compute instance from which you want to uninstall Storage Gateway.
2. Stop Storage Gateway:

```
sudo ocisg down
```

3. If the `ocisg_data` container exists in `docker ps -a output`, remove it:

```
sudo docker rm -v ocisg_data
```

4. Delete the image in `docker`:

```
sudo docker rmi $(sudo docker images| grep ocisg | awk '{print $3}')
```

5. Delete all the files in /usr/bin/ that begin with `ocisg`:

```
sudo rm /usr/bin/ocisg*
```

6. View the contents of the file `gateway_config`:

```
cat /etc/gateway_config
```

Sample output:

```
$ cat /etc/gateway_config

DATASTORAGE=/ocisg/cache
MDSTORAGE=/ocisg/metadata
LOGSTORAGE=/ocisg/log
PROXY=
USE_SSL=
MEMORY=
NETWORK=bridge
HTTP_FRAMEWORK=
ADMINPORT=443
NFSPORT=32769
RESTPORT=32768
```

5. Delete the `DATASTORAGE` directory, for example:

```
sudo rm -rf /ocisg/cache
```

6. Delete the `MDSTORAGE` directory , for example:

```
sudo rm -rf /ocisg/metadata
```

7. Delete the `LOGSTORAGE` directory , for example:

```
sudo rm -rf /ocisg/log
```

8. Delete the `gateway_config` file:

```
sudo rm /etc/gateway_config
```

9. Delete the Storage Gateway installation directory `ocisg`:

```
sudo rm -rf /opt/ocisg
```

Getting Help with Storage Gateway

This topic provides information about getting help with Oracle Cloud Infrastructure Storage Gateway.

Contacting Oracle Support

If you need technical support or help with Storage Gateway, you can go to My Oracle Support and create a service request. See Creating a Service Request for information.

Downloading the Support Bundle

If you contact Oracle Support about any issue with Storage Gateway, you might need to provide a support bundle to help the Oracle Support technicians diagnose the issue.

1. Log in to the management console.
2. Click **System** in the upper-right corner of the management console.
3. Click **Help**.
4. Click **Download Support Bundle** in **System Logs**.
   You can download and save the support bundle.

Contents of the Support Bundle

The support bundle contains the following information:

- All of the logs needed for diagnostics.
- Local storage usage information.
- Basic system information such as memory size, Docker version, and the Storage Gateway version.
- A list of file systems.
- Cloud Sync job details.

CHAPTER 26 Streaming Service Overview

The Oracle Cloud Infrastructure Streaming service provides a fully managed, scalable, and durable storage solution for ingesting continuous, high-volume streams of data that you can consume and process in real time. Streaming can be used for messaging, ingesting high-volume data such as application logs, operational telemetry, web click-stream data, or other use cases in which data is produced and processed continually and sequentially in a publish-subscribe messaging model.

Streaming Usage Scenarios

Here are some of the many possible uses for Streaming:

- **Metric and log ingestion**: Use streaming as an alternative for traditional file-scraping approaches to help make critical operational data more quickly available for indexing, analysis, and visualization.
- **Messaging**: Use streaming to decouple components of large systems. Streaming provides a pull/buffer-based communication model with sufficient capacity to flatten load spikes and the ability to feed multiple consumers with the same data independently. Key-scoped ordering and guaranteed durability provide reliable primitives to implement various messaging patterns, while high throughput potential allows for such a system to scale well.
- **Web/Mobile activity data ingestion**: Use streaming for capturing activity from websites or mobile apps (such as page views, searches, or other actions users may take). This information can be used for real-time monitoring and analytics, as well as in data warehousing systems for offline processing and reporting.
- **Infrastructure and apps event processing**: Use streaming as a unified entry point for cloud components to report their life cycle events for audit, accounting, and related activities.

Streaming Concepts

The following concepts are essential to working with Streaming.

STREAM

A partitioned, append-only log of messages.

PARTITION

A partition is a section of a stream. Partitions allow you to distribute a stream by splitting messages across multiple nodes. Each partition can be placed on a separate machine to allow for multiple consumers to read from a stream in parallel.

CURSOR

A pointer to a location in a stream. This location could be a pointer to a specific offset or time in a partition, or to a groups' current location.

MESSAGE

A Base64-encoded record that is published to a stream.

PRODUCER

An entity that publishes messages to a stream.

CONSUMER

An entity that reads messages from one or more streams.

CONSUMER GROUP

A consumer group is a set of instances which coordinates messages from all of the partitions in a stream. Instances in a consumer group maintain group membership through interaction; lack of interaction for a period of time results in a timeout, removing the instance from the group.

TOPIC/KEY

An identifier used to group related messages.

OFFSET

The location of a message within a partition. You can use the offset to restart reading from a stream.

PERMISSIONS

You can use IAM to set permissions on the following operations: list, get, update, create, and delete streams.

How Streaming Works

The Streaming service provides a robust, scalable mechanism that you can use to produce and consume high volumes of data between application components.

Here's how Streaming works: a *producer* publishes *messages* to a *stream*, which is an append-only log. These messages are distributed among the partitions using the message's key.

Streams are divided into a number of *partitions* for scalability. Partitions allow you to distribute a stream by splitting messages across multiple nodes (or brokers). Each partition can be placed on a separate machine to allow multiple consumers to read a stream in parallel. Multiple consumers can read from any partition regardless of where the partition is hosted.

A *consumer* can read messages from one or more streams. Each message within a stream is marked with an offset value, so a consumer can pick up where it left off if it is interrupted.

You can use the streaming service by:

- Creating a stream using the Console or API.
- Using a producer to publish data to the stream.

- Building consumers to read and process messages from a stream using the GetMessages API .

Limits on Streaming Resources

The Streaming service has the following limitations:

- Message retention of up to a maximum 7 days
- Throughput is limited to 1MB per second per partition
- Each partition can handle up to 1MB maximum message size
- Each partition can handle 5 GetMessages API calls per second
- Each partition can support up a maximum total data write rate of 1MB per second
- Each enterprise tenancy has a limit of 5 partitions (non-enterprise partitions have a limit of 0, but you can request more)

See Service Limits for a list of applicable limits and instructions for requesting a limit increase.

Resource Identifiers

Most types of Oracle Cloud Infrastructure resources have a unique, Oracle-assigned identifier called an Oracle Cloud ID (OCID). For information about the OCID format and other ways to identify your resources, see Resource Identifiers.

Ways to Access Oracle Cloud Infrastructure

You can access Oracle Cloud Infrastructure using the Console (a browser-based interface) or the REST API. Instructions for the Console and API are included in topics throughout this guide. For a list of available SDKs, see Software Development Kits and Command Line Interface.

To access the Console, you must use a supported browser. Oracle Cloud Infrastructure supports the latest desktop versions of Google Chrome, Microsoft Edge, Internet Explorer 11, Safari, Firefox, and Firefox ESR. Note that private browsing mode is not supported for Firefox, Internet Explorer, or Edge. Mobile browsers are not supported.

For general information about using the API, see REST APIs.

Using Streaming

To get started with Streaming, see the following topics:

- For instructions on how to manage streams, see Managing Streams.
- For information about publishing messages to a stream, see Publishing Messages.
- For information on how to consume messages, see Consuming Messages.
- For SDK information, see Oracle Cloud Infrastructure SDKs.

Authentication and Authorization

Each service in Oracle Cloud Infrastructure integrates with IAM for authentication and authorization, for all interfaces (the Console, SDK or CLI, and REST API).

An administrator in your organization needs to set up groups, compartments, and policies that control which users can access which services, which resources, and the type of access. For example, the policies control who can create new users, create and manage the cloud network, launch instances, create buckets, download objects, etc. For more information, see Getting Started with Policies. For specific details about writing policies for each of the different services, see Policy Reference.

If you're a regular user (not an administrator) who needs to use the Oracle Cloud Infrastructure resources that your company owns, contact your administrator to set up a user ID for you. The administrator can confirm which compartment or compartments you should be using.

For common policies used to authorize Streaming users, see Common Policies.

For in-depth information on granting users permissions for the Streaming service, see Details for the Streaming Service in the IAM policy reference.

Tagging Resources

You can apply tags to your resources to help you organize them according to your business needs. You can apply tags at the time you create a resource, or you can update the resource later with the desired tags. For general information about applying tags, see Resource Tags.

Last edited: 5/24/2019 8:11 AM

Managing Streams

This topic describes how to work with streams.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don't have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let streaming users manage streams lets the specified group do everything with streaming and related Streaming service resources.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for the Streaming service, see Details for the Streaming Service in the IAM policy reference.

Using the Console

To create a stream

1. Click the **Create Stream** button at the top of the topic list.
2. In the **Create Stream** dialog box, configure your topic:
    a. **Compartment** : Required. Select a compartment from the drop-down list.
    b. **Stream Name**: Required. Specify a friendly name for the topic. It does not have to be unique, but it cannot be changed. Avoid entering confidential information.
3. In the **Stream Settings** panel:
    a. **Retention (in Hours)**: Enter the number of hours (from 24 to 168) to retain messages. The default value is 24.
    b. **Number of Partitions:** Enter the number of partitions for the stream. The maximum number is based on the limits for your tenancy.
        i. You can optionally configure the stream settings based on estimated capacity. To do this, click on the **Configure stream settings based on estimated capacity** link to reveal the **Capacity Estimates** panel.
            i. Fill out the capacity estimates and then click **the Estimate Stream Size** button. This will automatically populate the **Number of partitions** field.
4. **Tags**: Optionally, you can apply tags. If you have permissions to create a resource, you also have permissions to apply free-form tags to that resource. To apply a defined tag, you must have permissions to use the tag namespace. For more information about tagging, see Resource Tags. If you are not sure if you should apply tags, skip this option (you can apply tags later) or ask your administrator.

5. Click **Submit**.

To delete a stream

1. Click the check box next to the topics you want to delete and then click **Delete Topic**.
2. Confirm when prompted.

To produce a test message

1. Click a stream to display the stream details page.
2. Click the **Produce Test Message** button.
3. On the **Test Stream** dialog, enter the text-only message to produce in the **Data** text box.
4. Click the **Produce** button.

To show recent messages on a stream

1. Click a stream to display the stream details page.
2. In the **Recent Messages** panel, click the **Refresh** button.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to manage streams:

- CreateStream
- ListStreams
- GetStream

- UpdateStream
- DeleteStream

Publishing Messages

This topic covers how to emit messages to a stream.

Overview

To publish messages:

- Create a stream using CreateStream.
- Publish a message to a specified stream using PutMessages.

Publishing Messages

Once a stream is created and active, you can publish messages.

A message is composed of a key and a value. Both the key and the value are byte arrays. If you don't explicitly supply a key, one is generated automatically.

The message is published to a partition in the stream. If there is more than one partition, the partition where the message is published is calculated using the message's key. If the key is null, the partition is calculated using a random 16-byte value.

For messages with a null key, do not expect messages with same value to go on the same partition, since the partitioning scheme may change. Passing a null key will put the message in a random partition. If you want to ensure that messages with the same value go to the same partition, you should use the same key for those messages.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let streaming users manage streams lets the specified group do everything with streaming and related Streaming service resources.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for databases, see Details for the Streaming Service in the IAM policy reference.

Using the Console

To publish a message

1. Click on the stream that you want to publish on.
2. Click the **Produce Test Message** button.
3. Type the message in the **Data** text box.
4. Click **Produce**.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to produce messages:

- PutMessages

Consuming Messages

This topic covers how to consume messages from a stream.

Overview

Consuming messages requires you to:

- Create a cursor
- Use the cursor to read messages

Using Cursors

A cursor is a pointer to a location in a stream. This location could be a pointer to a specific offset or time in a partition, or to a groups' current location.

To consume messages, use the CreateCursor or CreateGroupCursor API to create a cursor on a partition to indicate where to start consuming messages.

There are five supported cursor types:

- `TRIM_HORIZON` - Start consuming from the oldest available message in the stream. Create a cursor at the TRIM_HORIZON to consume all messages in a stream.
- `AT_OFFSET` - Start consuming at a specified offset. The offset must be greater than or equal to the offset of the oldest message and less than or equal to the latest published offset.
- `AFTER_OFFSET` - Start consuming after the given offset. This cursor has the same restrictions as the AT_OFFSET cursor.
- `AT_TIME` - Start consuming from a given time. The timestamp of the returned message will be on or after the supplied time.
- `LATEST` - Start consuming messages that were published after you created the cursor.

Consuming Messages

Once you've created a cursor, you can start to consume messages using GetMessages. Each call to GetMessages returns the cursor to use in the next GetMessages call. The returned cursor will never be null and expires in 5 minutes. As long as you keep consuming, you should never have to re-create a cursor.

Consuming Messages using a Consumer Group

Consumers can be configured to consume messages as part of a group. Stream partitions are distributed among members of a group, such that messages from any single partition are only sent to a single consumer.

Partition assignments are rebalanced as consumers join or leave the group. Group consumption is accomplished using the same cursor mechanism as with single consumers, but using a different kind of cursor.

To create a consumer, create a group cursor, providing a group, instance name, and cursor type. Groups are created on the first request to create a cursor; their retention period is the same as their assigned stream:

```
CreateGroupCursorRequest groupRequest = CreateGroupCursorRequest.builder()
                .streamId(streamId)
                .createGroupCursorDetails(CreateGroupCursorDetails.builder()
                        .groupName(groupName)
                        .instanceName(instanceName)
                        .type(CreateGroupCursorDetails.Type.TrimHorizon)
                        .commitOnGet(true)
                        .build())
                .build());

CreateGroupCursorResponse groupCursorResponse = streamClient.createGroupCursor(groupRequest);
String groupCursor = groupCursorResponse.getCursor().getValue();
// this groupCursor can be used in the same message loop a described above; subsequent getMessages calls
return an updated groupCursor.
```

Once you've created a cursor, you can start to consume messages using GetMessages. Each call to GetMessages returns the cursor to use in the next GetMessages call. The returned cursor will never be null and expires in 5 minutes. As long as you keep consuming, you should never have to re-create a cursor.

Required IAM Policy

To use Oracle Cloud Infrastructure, you must be given the required type of access in a policy written by an administrator, whether you're using the Console or the REST API with an SDK, CLI, or other tool. If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in.

For administrators: The policy in Let streaming users manage streams lets the specified group do everything with streaming and related Streaming service resources.

If you're new to policies, see Getting Started with Policies and Common Policies. If you want to dig deeper into writing policies for databases, see Details for the Streaming Service in the IAM policy reference.

Using the Console

You cannot use the console to consume messages.

Using the API

For information about using the API and signing requests, see REST APIs and Security Credentials. For information about SDKs, see Software Development Kits and Command Line Interface.

Use these API operations to consume messages:

- CreateCursor
- GetMessages

Using the Streaming SDK

This topic covers how to use the Streaming SDK.

Getting Started

For information on installing and configuring the Oracle Cloud Infrastructure SDKs, see Developer Tools.

Architecture Overview

Streaming contains the following key building blocks:

- **Stream:** A partitioned, append-only log of messages.
- **Partition:** A section of a stream. Partitions allow you to distribute a stream by splitting messages across multiple nodes. Each partition can be placed on a separate machine to allow for multiple consumers to read from a topic in parallel.
- **Producer:** An entity that publishes messages to a stream.
- **Consumer:** An entity that reads messages from a stream.
- **Consumer group:** A group of consumers that can read independently read messages from separate partitions of a stream.

Streaming Clients

The Streaming SDK is encapsulated in two clients: the `StreamAdminClient` and the `StreamClient`.

The `StreamAdminClient` incorporates the control plane operations of the streaming service. You can use it to create, delete, update, modify, and list streams.

To instantiate the `StreamAdminClient` object:

```
StreamAdminClient adminClient = new StreamAdminClient([authProvider]);
adminClient.setEndpoint("https://streaming.r2.oracleiaas.com"); // You cannot use the setRegion method
```

The `StreamClient` is used to publish and consume messages.

To instantiate a `StreamClient` object:

```
// First you have to get the stream you want to consume/publish.
// You can either make a CreateStream, GetStream, or ListStream call. They all return a
"messagesEndpoint" as part of a Stream object.
// That endpoint NEEDS to be used when creating the StreamClient object.
GetStreamRequest getStreamRequest = GetStreamRequest.builder().streamId(streamId).build();
Stream stream = adminClient.getStream(getStreamRequest).getStream();

StreamClient streamClient = new StreamClient([authProvider]);
streamClient.setEndpoint(stream.getMessagesEndpoint());
```

Creating a Stream

To create a stream, use the `createStream` method of `StreamAdminClient`. Creating a stream is an asynchronous operation. You can check on the completion of the create operation by checking that the `lifecycleStateDetails` property of your new stream is either `Active` or `Failed`.

The following is an example showing how to create a stream:

```
// No error handling
CreateStreamDetails createStreamDetails =  CreateStreamDetails.builder()
   .partitions(5) // number of partitions you want in your stream

   .name("myStream") // the name of the stream - only used in the console
   .compartmentId(tenancy) // the compartment id you want your stream to live in
   .build();

// You can also add tags to the createStreamDetails object.
CreateStreamRequest createStreamRequest =
   CreateStreamRequest.builder()
     .createStreamDetails(createStreamDetails)
     .build();

Stream stream = adminClient.createStream(createStreamRequest).getStream();

while (stream.getLifecycleState() != Stream.LifecycleState.Active && stream.getLifecycleState() !=
Stream.LifecycleState.Failed) {

   GetStreamRequest getStreamRequest = GetStreamRequest.builder().streamId(stream.getId()).build();
   stream = adminClient.getStream(getStreamRequest).getStream();
}

// Handle stream Failure
```

Deleting a Stream

To delete a stream, use the `deleteStream` method API of the `StreamAdminClient`. Deleting a stream is an asynchronous operation; the stream state changes to `Deleted` once the delete operation is finished. During the deletion process, the stream can't be used for consuming or producing messages.

The following example shows how to use the `deleteStream` method to delete a stream:

```
// No error handling
DeleteStreamRequest deleteStreamRequest =

    DeleteStreamRequest.builder()

      .streamId(stream.getId())
       .build();

adminClient.deleteStream(deleteStreamRequest);
```

Listing Streams

Use the `listStreams` method to return a list of streams for a given compartment.

You can filter the returned list by OCID, life cycle state, and name.

The results can be sorted in ascending or descending order by name or creation time.

The results are passed back in a paginated list. A token is passed back with each page of results; pass this token back to the `getOpcNextPage` method to retrieve the next page of results. A null token returned from `getOpcNextPage` indicates that no more results are available.

For example:

```
// No error handling
ListStreamsRequest listStreamsRequest =
        ListStreamsRequest.builder()
                .compartmentId(tenancy)
                .build();
// You can filter by OCID (exact match only) [builder].id(streamId) -> This will return 0..1 item
// You can filter by name (exact match only) [builder].name(name) -> This will return 0..n items
// You can order the result per TimeCreated or Name [builder].sortBy(SortBy.[TimeCreated|Name])
// You can change the ordering [builder].sortOrder(SortOrder.[Asc|Desc])
// You can filter by lifecycleState [builder].lifecycleState(lifecycleState)

String page;
do {
    ListStreamsResponse listStreamsResponse = adminClient.listStreams(listStreamsRequest);
    List<StreamSummary> streams = listStreamsResponse.getItems();
    // Do something with the streams
    page = listStreamsResponse.getOpcNextPage();
} while (page != null);
```

Retrieving Stream Details

To get details about a stream, use the `getStream` method and then examine the properties of the stream. For example:

```
// No error handling
GetStreamRequest getStreamRequest =
        GetStreamRequest.builder()
                .streamId(streamId)
                .build();

Stream stream = adminClient.getStream(getStreamRequest).getStream();
```

Publishing Messages

Once a stream is created and active, you can publish messages using the `streamClient.putMessages` method.

A message is composed of a key (which can be null) and a value. Both key and value are byte arrays.

The message is published to a partition in the stream. If there is more than one partition, the partition where the message is published is calculated using the message's key. If the key is null, the partition is calculated using a subset of the value. For messages with a null key, do not expect messages with same value to go on the same partition since the partitioning scheme may change. Sending a null key will put the message in a random partition. If you want to ensure that messages with the same value go to the same partition, you should use the same key for those messages.

The following code shows how to publish a message:

```
// No error handling
List<PutMessagesDetailsEntry> messages = new ArrayList<>();

for (int i = 0; i < 40; i++) {
    byte[] key = "myKey".getBytes(Charsets.UTF_8); // In that case, all messages will go on the same
partition since the key is the same.
    byte[] value = UUID.randomUUID().toString().getBytes(Charsets.UTF_8);
    messages.add(new PutMessagesDetailsEntry(key, value));
}
```

```
PutMessagesDetails putMessagesDetails =
        PutMessagesDetails.builder()
                .messages(messages)
                .build();

PutMessagesRequest putMessagesRequest =
        PutMessagesRequest.builder()
                .putMessagesDetails(putMessagesDetails)
                .build();

PutMessagesResult putMessagesResult = streamClient.putMessages(putMessagesRequest).getPutMessagesResult
();
// It's not because the call didn't fail that the messages were successfully published!
int failures = putMessagesResult.getFailures();
// If failures is > 0, it means we have a partial-success call.
List<PutMessagesResultEntry> entries = putMessagesResult.getEntries();
// entries is a list of the same size as the list of messages you sent.
// It is guaranteed that the order of the messages is the same as when you sent them.
// Each entry contains either "offset/partition/timestamp" if the message was successfully published
// or "error/errorMessage" if it failed.
if (failures != 0) {
    entries.forEach(entry -> {
        if (StringUtils.isNotEmpty(entry.getError())) {
            // That particular message failed to get published.
            // It could be a throttle error and in that case error would be "429" and errorMessage would
contain a meaningful message.
            // Or it could be an internal error on our side and error would be "500".

            // Possible solution would be to republish only failed messages.
        }
    });
}
```

Consuming Messages

Consuming messages requires the use of a cursor, which is a pointer to an offset into a partition.

There are five supported cursor types:

- `TRIM_HORIZON` - Start consuming from the oldest available message in the stream. Create a cursor at the TRIM_HORIZON to consume all messages in a stream.
- `AT_OFFSET` - Start consuming at a specified offset. The offset must be greater than or equal to the offset of the oldest message and less than or equal to the latest published offset.
- `AFTER_OFFSET` - Start consuming after the given offset. This cursor has the same restrictions as the AT_OFFSET cursor.
- `AT_TIME` - Start consuming from a given time. The timestamp of the returned message will be on or after the supplied time.
- `LATEST` - Start consuming messages that were published after you created the cursor.

To create a `TRIM_HORIZON` cursor, which starts consuming starting from the oldest available message:

```
// No error handling
CreateCursorDetails createCursorDetails =
        CreateCursorDetails.builder()
                .type(Type.TrimHorizon)
                .partition("0")
                .build();
// If using AT_OFFSET or AFTER_OFFSET you need to specify the offset [builder].offset(offset)
// If using AT_TIME you need to specify the time [builder].time(new Date(xxx))

CreateCursorRequest createCursorRequest =
        CreateCursorRequest.builder()
                .createCursorDetails(createCursorDetails)
                .build();

String cursor = streamClient.createCursor(createCursorRequest).getCursor().getValue();
// Cursor will then be used to get messages from the stream.
```

Once you've created a cursor, you can start to consume messages using the GetMessages method. Each call to GetMessages returns the cursor to use in the next GetMessages call. The returned cursor is not null and expires in 5 minutes. As long as you keep consuming, you should never have to re-create a cursor.

Here's an example of using a cursor to retrieve messages:

```
// No error handling (there is a high chance of getting a throttling error using a tight loop)
while (true) { // or your own exit condition
    GetMessagesRequest getMessagesRequest =
            GetMessagesRequest.builder()
                    .cursor(cursor)
                    .build();

    GetMessagesResponse getMessagesResponse = streamClient.getMessages(getMessagesRequest);

    // This could be empty, but we will always return an updated cursor
    getMessagesResponse.getItems().forEach(message -> {
        // Process the message
    });

    cursor = getMessagesResponse.getOpcNextCursor();Consuming Messages
```

Using Consumer Groups

Consumers can be configured to consume messages as part of a group. Stream partitions are distributed among members of a group, such that messages from any single partition are only sent to a single consumer.

Partition assignments are rebalanced as consumers join or leave the group. Group consumption is accomplished using the same cursor mechanism as with single consumers, but using a different kind of cursor.

How Consumer Groups Work

A consumer group is a set of instances which coordinate to consume messages from all of the partitions in a stream. Instances maintain membership to a group through interaction; lack of interaction for a period of time results in a timeout, removing the instance from the group. Partitions are reserved for specific instances in a group; reservations are rebalanced in response to specific group events, such as an instance joining the group, or instance time-out.

While an instance maintains membership to a group, calls to get messages will return messages from only partitions reserved for that instance. Partition reservation attempts to ensure that messages are only processed by a single instance.

The set of instances in a group is expected to change over time; transience can occur for many reasons, commonly because of server failure, scaling patterns, or operational management.

Persisting consumer processing state beyond the lifetime of an instance, allows instances to come and go, picking up where appropriate in context of the group.

An instance commits offsets of processed messages to ensure that they are not processed again by other instances in the same group.

Consuming Messages with a Consumer Group

To create a consumer group, create a group cursor, providing a group, instance name, and cursor type. Groups are created on the first request to create a cursor; their retention period is the same as their assigned stream:

```
CreateGroupCursorRequest groupRequest = CreateGroupCursorRequest.builder()
                .streamId(streamId)
                .createGroupCursorDetails(CreateGroupCursorDetails.builder()
                        .groupName(groupName)
                        .instanceName(instanceName)
                        .type(CreateGroupCursorDetails.Type.TrimHorizon)
                        .commitOnGet(true)
                        .build())
                .build());

CreateGroupCursorResponse groupCursorResponse = streamClient.createGroupCursor(groupRequest);
String groupCursor = groupCursorResponse.getCursor().getValue();
// this groupCursor can be used in the same message loop a described above; subsequent getMessages calls
return an updated groupCursor.
```

Once you've created a cursor, you can start to consume messages using GetMessages. Each call to GetMessages returns the cursor to use in the next GetMessages call. The returned cursor is never null and expires in 5 minutes. As long as you keep consuming, you should never have to re-create a cursor.

Streaming Metrics

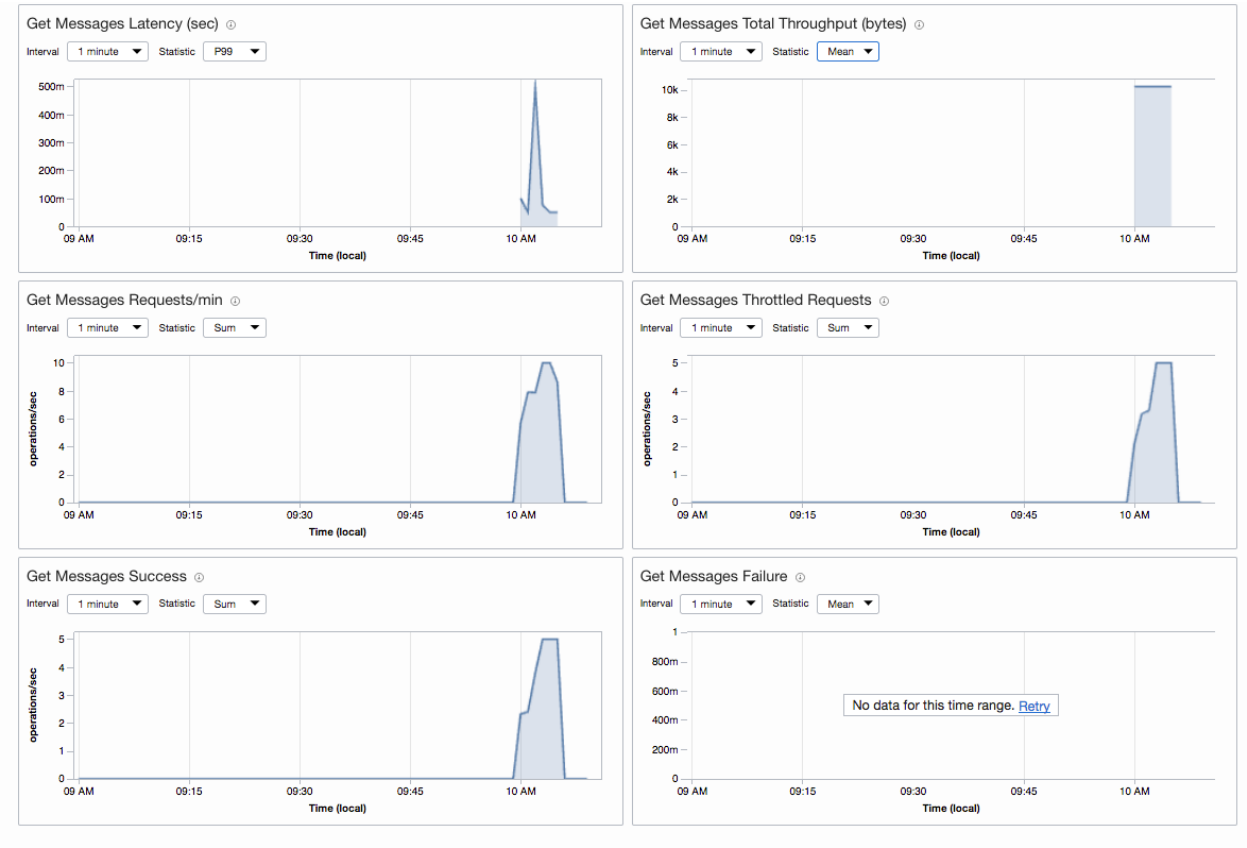
This topic describes the metrics emitted by the Streaming service using the metric namespace `oci_oss`.

Overview of Streaming Metrics

The Streaming service provides metrics showing how the service is performing. These metrics are automatically available.

You can use these metrics to:

- Understand the produce/consume latency for a real-time application.
- Calculate and validate the price of service usage.
- Monitor changes in throughput over time.
- Check the time that the last message was consumed.



To view a default set of metrics charts in the Console, navigate to the Service Metrics page and then select the `oci_oss` metric namespace.

Terminology

The following terms are used when discussing Streaming service metrics:

- *Namespace:* A namespace is a container for Streaming metrics. The namespace identifies the application or service sending the metrics. The namespace for the Streaming is `oci/oss`.
- *Metrics:* Metrics are the fundamental concept in telemetry and monitoring. Metrics define a time-series set of datapoints. Each metric is uniquely defined by namespace,

metric name, compartment identifier, and a set of one or more dimensions, and a unit of measure. Each datapoint has a time stamp, a value, and a count associated with it.

- *Dimensions:* A dimension is a key-value pair that defines the characteristics associated with the metric; for ecample, `resourceId(streamOcid`.
- *Statistics :* Statistics are metric data aggregations over specified periods of time. Aggregations are done using the namespace, metric name, dimensions, and the data point unit of measure within the time period specified.
- *Alarms:* Alarms are used to automate operations monitoring and performance. An alarm keeps track of changes that occur over a specific period of time and performs one or more defined actions, based on the rules defined for the metric.

Available Metrics

The following tables describe the available Streaming metrics.

Producers

Metric	Metric Display Name	Unit	Description	Dimensions
PutMessagesLatency.Time	Put Messages Latency	time (ms)	Time taken for put messages operation measured over time range	streamOcid,regionId
PutMessagesThroughput.Bytes	Put Messages Total Throughput	bytes	Bytes pushed to the stream measured over time	
PutMessagesThroughput.Count	Put Messages Records/sec	count	Count of messages pushed to stream measured over time	
PutMessagesThrottling.Count	Put Messages Throttled Records/sec	count	Number of put messages throttled either due to volume or requests measured over time	

Metric	Metric Display Name	Unit	Description	Dimensions
PutMessagesSuccess.Count	Put Messages Success	count	Successful Requests for put messages per stream measured over time	
PutMessagesFault.Count	Put Messages Failure	count	Total Failed putMessage Requests per stream measured over time	
PutMessagesRecords.Count	Put Messages Requests	count	Number of records published to a stream measured over time	
PutMessages.Bytes	Put Messages Bytes	bytes	Bytes pushed to a stream over time	
PutMessages.Count	Put Messages Count	count	Number of messages pushed over time	

Consumers

Metric	Metric Display Name	Unit	Description	Dimensions
GetMessagesLatency.Time	Get Messages Latency	time (ms)	Time taken for get messages operation measured over time range	streamOcid,regionId
GetMessagesThroughput.Bytes	Get Messages Total Throughput	bytes	Bytes retrieved from stream measured over time	
GetMessagesThroughput.Count	Get Messages Requests/sec	count	Count of messages read from stream measured over time	
GetMessagesThrottling.Count	Get Messages Throttled Requests	count	Number of get messages throttled either due to volume or requests measured over time	

Metric	Metric Display Name	Unit	Description	Dimensions
GetMessagesSuccess.Count	Get Messages Success	count	Successful Requests for get messages per stream measured over time	
GetMessagesFault.Count	Get Messages Failure	count	Total Failed getMessage Requests per stream measured over time	
GetMessages.Bytes	Get Messages Bytes	bytes	Bytes retrieved from a stream over time	
GetMessages.Count	Get Messages Count	count	Number of messages read over time	

Consumer Groups

Metric	Metric Display Name	Unit	Description	Dimensions
`totalActiveConsumers.Count`	Total Active Consumer Groups	count	Number of active consumers.	streamOcid,regionId
`totalActiveConsumersGroups.Count`	Total Active Consumers	count	Number of active consumer groups.	

CHAPTER 27 Developer Tools

This chapter includes general information about using the Oracle Cloud Infrastructure REST API and developer tools.

Software Development Kits and Command Line Interface

Oracle Cloud Infrastructure provides a number of Software Development Kits (SDKs) and a Command Line Interface (CLI) to facilitate development of custom solutions.

- Software Development Kits (SDKs)
  Build and deploy apps that integrate with Oracle Cloud Infrastructure services. Each SDK provides the tools you need to develop an app, including code samples and documentation to create, test, and troubleshoot. In addition, if you want to contribute to the development of the SDKs, they are all open source and available on GitHub.
  - SDK for Java
  - Python SDK
  - Ruby SDK
  - Go SDK
- Command Line Interface (CLI)
  The CLI provides the same core capabilities as the Oracle Cloud Infrastructure Console, plus additional commands that can extend the Console's functionality. Convenient for developers or anyone who prefers the command line to a GUI.

SDK for Java

The SDK for Java enables you to write code to manage Oracle Cloud Infrastructure resources.

This SDK and sample is dual-licensed under the Universal Permissive License 1.0 and the Apache License 2.0; third-party content is separately licensed as described in the code.

Download: GitHub

Requirements

To use the SDK for Java, you must have the following:

- An Oracle Cloud Infrastructure account.
- A user created in that account, in a group with a policy that grants the desired permissions. This can be a user for yourself, or another person/system that needs to call the API. For an example of how to set up a new user, group, compartment, and policy, see Adding Users. For a list of typical policies you may want to use, see Common Policies.
- A key pair used for signing API requests, with the public key uploaded to Oracle. Only the user calling the API should be in possession of the private key. For more information, see Configuring Credentials.
- Java 8
- A TTL value of 60. For more information, see Java Virtual Machine TTL for DNS Name Lookups.

Services Supported

- Announcements
- Audit
- Budgets
- Container Engine for Kubernetes
- Compute Autoscaling
- Core Services (Networking, Compute, Block Volume)
- Database
- DNS
- Email Delivery
- File Storage
- Health Checks
- IAM

- Key Management
- Load Balancing
- Monitoring
- Notifications
- Object Storage
- Resource Manager
- Search
- Streaming
- Web Application Acceleration and Security

Contact Us

CONTRIBUTIONS

Got a fix for a bug or a new feature you'd like to contribute? The SDK is open source and accepting pull requests on GitHub.

NOTIFICATIONS

To be notified when a new version of the SDK for Java is released, subscribe to the Atom feed.

QUESTIONS OR FEEDBACK

- GitHub Issues: To file bugs and feature requests only
- Stack Overflow: Please use the oracle-cloud-infrastructure and oci-java-sdk tags in your post
- Developer Tools section of the Oracle Cloud forums
- My Oracle Support

Getting Started

This topic describes how to install and configure the Oracle Cloud Infrastructure SDK for Java.

DOWNLOADING THE SDK

You can download the SDK for Java as a zip archive from GitHub. It contains the SDK, all of its dependencies, documentation, and examples. For best compatibility and to avoid issues, use the version of the dependencies included in the archive. Some notable issues are:

- Bouncy Castle: The SDK bundles 1.60, but if you need FIPS compliance, you must download and use a FIPS-certified version. The SDK supports bc-fips 1.0.1 and bcpkix-fips 1.0.1. You can download them at: https://www.bouncycastle.org/fips-java/
- Jersey Core and Client: The SDK bundles 2.24.1, which is required to support large object uploads to Object Storage. Older versions will not support uploads greater than ~2.1 GB.
- Jax-RS API: The SDK bundles 2.0.1 of the spec. Older versions will cause issues.



Note

The SDK for Java is bundled with Jersey, but you can also use your own JAX-RS implementation. For details, see Using Your Own JAX-RS Implementation

CONFIGURING THE SDK

The SDK services need two types of configuration: credentials and client-side HTTP settings.

CONFIGURING CREDENTIALS

First, you need to set up your credentials and config file. For instructions, see SDK and CLI Configuration File.

Next you need to set up the client to use the credentials. The credentials are abstracted through an `AuthenticationDetailsProvider` interface. Clients can implement this however you choose. We have included a simple POJO/builder class to help with this task (`SimpleAuthenticationDetailsProvider`).

- You can load a config with or without a profile:

```
ConfigFile config
    = ConfigFileReader.parse("~/.oci/config");
ConfigFile configWithProfile
    = ConfigFileReader.parse("~/.oci/config", "DEFAULT");
```

- The private key supplier can be created with the file path directly, or using the config file:

```
Supplier<InputStream> privateKeySupplier
    = new SimplePrivateKeySupplier("~/.oci/oci_api_key.pem");
Supplier<InputStream> privateKeySupplierFromConfigEntry
    = new SimplePrivateKeySupplier(config.get("key_file"));
```

- To create an auth provider using the builder:

```
AuthenticationDetailsProvider provider
    = SimpleAuthenticationDetailsProvider.builder()
        .tenantId("myTenantId")
        .userId("myUserId")
        .fingerprint("myFingerprint")
        .privateKeySupplier(privateKeySupplier)
        .build();
```

- To create an auth provider using the builder with a config file:

```
AuthenticationDetailsProvider provider
    = SimpleAuthenticationDetailsProvider.builder()
        .tenantId(config.get("tenancy"))
        .userId(config.get("user"))
        .fingerprint(config.get("fingerprint"))
        .privateKeySupplier(privateKeySupplier)
        .build();
```

- Finally, if you use standard config file keys and the standard config file location, you can simplify this further by using `ConfigFileAuthenticationDetailsProvider`:

```
AuthenticationDetailsProvider provider
    = new ConfigFileAuthenticationDetailsProvider("ADMIN_USER");
```

CONFIGURING CLIENT-SIDE OPTIONS

Create a client-side configuration through the `ClientConfiguration` class. If you do not provide your own configuration, the SDK for Java uses a default configuration. To provide your own configuration, use the following:

```
ClientConfiguration clientConfig
    = ClientConfiguration.builder()
        .connectionTimeoutMillis(3000)
        .readTimeoutMillis(60000)
        .build();
```

After you have both a credential configuration and the optional client configuration, you can start creating service instances.

CONFIGURING CUSTOM OPTIONS

In the config file, you can insert custom key-value pairs that you define, and then reference them as necessary. For example, you could specify a frequently used compartment ID in the config file like so (highlighted in red italics):

```
[DEFAULT]
user=ocid1.user.oc1..aaaaaaaat5nvwcna5j6aqzjcmdy5eqbb6qt2jvpkanghtgdaqedqw3rynjq
fingerprint=20:3b:97:13:55:1c:5b:0d:d3:37:d8:50:4e:c5:3a:34
key_file=~/.oci/oci_api_key.pem
tenancy=ocid1.tenancy.oc1..aaaaaaaaba3pv6wkcr4jqae5f15p2bcmdyt2j6rx32uzr4h25vqstifsfdsq
custom_compartment_
id=ocid1.compartment.oc1..aaaaaaaayzfqeibduyox6iib3olcmdar3ugly4fmameq4h7lcdlihrvur7xq
```

Then you can retrieve the value like so:

```
ConfigFile config
    = ConfigFileReader.parse("~/.oci/config");

String compartmentId = config.get("custom_compartment_id");
```

Configuration

This topic provides details on compatibility, advanced configurations, and add-ons for the Oracle Cloud Infrastructure SDK for Java.

SECURITY MANAGER PERMISSIONS

If your application needs to run inside the Java Security Manager, you must grant additional permissions by updating a policy file, or by specifying an additional or a different policy file at runtime.

The SDK requires the following permissions:

- Required by Jersey:

```
permission java.lang.RuntimePermission "getClassLoader";
permission java.lang.reflect.ReflectPermission "suppressAccessChecks";
permission java.lang.RuntimePermission "accessDeclaredMembers";
permission java.util.PropertyPermission "*", "read,write";
permission java.lang.RuntimePermission "setFactory";
```

- Required by the SDK to overwrite reserved headers:

```
permission java.util.PropertyPermission "sun.net.http.allowRestrictedHeaders", "write";
```

- Required by the SDK to open socket connections:

```
permission java.net.SocketPermission "*", "connect";
```

To include another policy file, in addition to Java Runtime Environment's default policy file, launch the Java Virtual Machine with:

```
java -Djava.security.manager -Djava.security.policy=</path/to/other_policy>
```

To replace the default policy file, launch the Java Virtual Machine with:

```
java -Djava.security.manager -Djava.security.policy==</path/to/other_policy>
```



Note

Use a single equals sign (=) when supplying an additional policy file. Use a double equals sign (==) only if you wish to replace the default policy file.

Java Virtual Machine TTL for DNS Name Lookups

The Java Virtual Machine (JVM) caches DNS responses from lookups for a set amount of time, called *time-to-live* (TTL). This ensures faster response time in code that requires frequent name resolution.

The JVM uses the networkaddress.cache.ttl property to specify the caching policy for DNS name lookups. The value is an integer that represents the number of seconds to cache the successful lookup. The default value for many JVMs, `-1`, indicates that the lookup should be cached forever.

Because resources in Oracle Cloud Infrastructure use DNS names that can change, we recommend that you change the the TTL value to 60 seconds. This ensures that the new IP address for the resource is returned on next DNS query. You can change this value globally or specifically for your application:

- To set TTL globally for all applications using the JVM, add the following in the `$JAVA_HOME/jre/lib/security/java.security` file:

  ```
  networkaddress.cache.ttl=60
  ```

- To set TTL only for your application, set the following in your application's initialization code:

  ```
  java.security.Security.setProperty("networkaddress.cache.ttl" , "60");
  ```

Apache Connector Add-On

The `oci-java-sdk-addons-apache` is an optional add-on to the SDK for Java that allows for configuring a client connection pool and an HTTP proxy. The add-on leverages the Jersey `ApacheConnectorProvider` instead of the SDK's default `HttpUrlConnectorProvider` when making service calls. The add-on can be found in the `bmc-addons` directory of the SDK.

For details on installation and configuration, see the Readme for the add-on.

Using Your Own JAX-RS Implementation

The SDK for Java is bundled with Jersey, but you can also use your own JAX-RS implementation. For an example of how to configure your own implementation, see RESTEasy Client Configurator Add-On and accompanying code samples.

RESTEasy Client Configurator Add-On

The `oci-java-sdk-addons-resteasy-client-configurator` is provided to demonstrate how to configure an alternate JAX-RS implementation. The add-on can be found in the `bmc-addons` directory of the SDK.

For details on installation and configuration, see the Readme for the add-on.

For code samples that demonstrate how to configure the client, see:

- ResteasyClientExample.java
- ResteasyClientWithObjectStorageExample.java
- InstancePrincipalsAuthenticationDetailsProviderWithResteasyClientExample.java

Using SLF4J for Logging

Logging in the SDK is done through SLF4J. SLF4J is a logging abstraction that allows the use of a user-supplied logging library (e.g., log4j). For more information, see the SLF4J manual.

The following is an example that enables basic logging to standard out. More advanced logging options can be configured by using the log4j binding.

1. Download the SLF4J Simple binding jar: SLF4J Simple Binding
2. Add the jar to your classpath (e.g., add it to the `/third-party/lib` directory of the SDK download)
3. Add the following VM arg to enable debug level logging (by default, info level is used): `-Dorg.slf4j.simpleLogger.defaultLogLevel=debug`

Concepts

This topic explains some of the key concepts for using the Oracle Cloud Infrastructure SDK for Java.

Synchronous Calls

To make synchronous calls, create an instance of the synchronous client. The general pattern for synchronous clients is that for a service named Example, there will be an interface named

ExampleService, and the synchronous client implementation will be called ExampleServiceClient. Here's an example of creating an Object Storage client:

```
AuthenticationDetailsProvider provider = ...;
ObjectStorage clientWithDefaultClientConfig = new ObjectStorageClient(provider);
clientWithDefaultClientConfig.setRegion(Region.US_ASHBURN_1);

ClientConfiguration clientConfig = ...;
ObjectStorage clientWithExplicitClientConfig = new ObjectStorageClient(provider, clientConfig);
clientWithExplicitClientConfig.setRegion(Region.US_ASHBURN_1);
```

Synchronous calls will block until the response is available. All SDK APIs return a response object (regardless of whether or not the API sends any content back). The response object typically contains at least a request ID that you can use when contacting Oracle support for help on a particular request.

```
ObjectStorage client = ...;
GetBucketResponse response = client.getBucket(
    GetBucketRequest.builder().namespaceName("myNamespace").bucketName("myBucket").build());
String requestId = response.getOpcRequestId();
Bucket bucket = response.getBucket();
System.out.println(requestId);
System.out.println(bucket.getName());
```

ASYNCHRONOUS CALLS

To make asynchronous calls, create an instance of the asynchronous client. The general pattern for asynchronous clients is that for a service named Example, there will be an interface named ExampleServiceAsync, and the asynchronous client implementation will be called ExampleServiceAsyncClient. Here's an example of creating an Object Storage client:

```
AuthenticationDetailsProvider provider = ...;
ObjectStorageAsync clientWithDefaultClientConfig = new ObjectStorageAsyncClient(provider);
clientWithDefaultClientConfig.setRegion(Region.US_ASHBURN_1);

ClientConfiguration clientConfig = ...;
ObjectStorageAsync clientWithExplicitClientConfig = new ObjectStorageAsyncClient(provider,
clientConfig);
clientWithExplicitClientConfig.setRegion(Region.US_ASHBURN_1);
```

Asynchronous calls will return immediately. You need to provide an AsyncHandler that will be invoked after the call completes either successfully or unsuccessfully:

```
ObjectStorageAsync client = ...;

AsyncHandler<GetBucketRequest, GetBucketResponse> handler = new AsyncHandler<GetBucketRequest,
GetBucketResponse>() {
        @Override
        public void onSuccess(GetBucketRequest request, GetBucketResponse response) {
            String requestId = response.getOpcRequestId();
            Bucket bucket = response.getBucket();
            System.out.println(requestId);
            System.out.println(bucket.getName());
        }

        @Override
        public void onError(GetBucketRequest request, Throwable error) {
            error.printStackTrace();
        }
};

Future<GetBucketResponse> future = client.getBucket(
        GetBucketRequest.builder().namespaceName("myNamespace").bucketName("myBucket").build(),
        handler);
```

POLLING WITH WAITERS

The SDK offers waiters that allow your code to wait until a specific resource reaches a desired state. A waiter can be invoked in both a blocking or a non-blocking (with asychronous callback) manner, and will wait until either the desired state is reached or a timeout is exceeded. Waiters abstract the polling logic you would otherwise have to write into an easy-to-use single method call.

Waiters are obtained through the service client (`client.getWaiters()`). Both a `Get<Resource>Request` and the desired lifecycle state are passed in to the `waiters.for<Resource>` method. For example:

```
public static Instance waitForInstanceProvisioningToComplete(  ComputeClient computeClient, String
instanceId) throws Exception {

    ComputeWaiters waiters = computeClient.getWaiters();
```

```
    GetInstanceResponse response = waiters.forInstance(
        GetInstanceRequest.builder().instanceId(instanceId).build(),
        Instance.LifecycleState.Running)
    .execute();

    return response.getInstance();
}
```

Each `waiters.for<Resource>` method has two versions:

- One version uses the default polling values. For example:

  ```
  waiters.forInstance(GetInstanceRequest, LifecycleState)
  ```

- The other version gives you full control over how long to wait and how much time between polling attempts. For example:

  ```
  waiters.forInstance(GetInstanceRequest, LifecycleState, TerminationStrategy, DelayStrategy)
  ```

THREADING MODEL

A client becomes thread-safe when it is initialized. After setting its endpoint, you can safely use a client in multiple threads and concurrently call methods on it.

You can reuse a client on multiple requests, both across concurrent threads or within a single thread. Unless the environment's resources are constrained, you should only close the client immediately before it goes out of scope.



Note

This guarantee applies only to the default JAX-RS implementation, Jersey. When using an alternate implementation, you must manage thread safety yourself. For more information, see Using Your Own JAX-RS Implementation

Uploading Large Objects

The Object Storage service supports multipart uploads to make large object uploads easier by splitting the large object into parts. The SDK for Java supports raw multipart upload operations for advanced use cases, as well as a higher level upload class that uses the multipart upload APIs. Managing Multipart Uploads provides links to the APIs used for multipart upload operations. Higher level multipart uploads are implemented using the UploadManager, which will: split a large object into parts for you, upload the parts in parallel, and then recombine and commit the parts as a single object in storage.

The UploadObject example shows how to use the UploadManager to automatically split an object into parts for upload to simplify interaction with the Object Storage service.

Raw Requests

Raw requests are useful, and in some cases necessary. Typical use cases include using your own HTTP client, making a OCI-authenticated request to an alternate endpoint, and making a request to a OCI API that is not currently supported in the SDK. The SDK for Java exposes the `RequestSigningFilter` class that you can use for signing non-standard requests.

The Raw Request example on GitHub shows how to:

- create a request signing filter
- integrate with an HTTP client (Jersey in this example) to authenticate requests

Setting the Endpoints

Service endpoints can be set in one of two ways.

- Call `setEndpoint()` on the service instance. This lets you specify a full host name (for example, https://www.example.com).
- Call `setRegion()` on the service instance. This selects the appropriate host name for the service for the given region. However, if the service is not supported in the region you set, the SDK for Java returns an error.

Note that a service instance cannot be used to communicate with different regions. If you need to make requests to different regions, create multiple service instances.

FORWARD COMPATIBILITY AND ENUMS

If you have conditional logic based on an enum, be sure that your code handles the `UnknownEnumValue` case to ensure forward compatibility. Some response fields are of type enum, but in the future, individual services may return values not covered by existing enums for that field. To address this possibility, every response field of type enum has an additional value named `UnknownEnumValue`. If a service returns a value that is not recognized by your version of the SDK, then the response field will be set to this value.

NEW REGION SUPPORT

If you are using a version of the SDK released prior to the announcement of a new region, you can use a workaround to reach it.

A *region* is a localized geographic area. For more information on regions and how to identify them, see Regions and Availability Domains.

A *realm* is a set of regions that share entities. You can identify your realm by looking at the domain name at the end of the network address. For example, the realm for `xyz.abc.123.oraclecloud.com` is `oraclecloud.com`.



Note

For the following code samples, be sure to supply the appropriate endpoints for your region.

ORACLECLOUD.COM REALM

For regions in the `oraclecloud.com` realm, you cannot pass new region names just as you would pass ones that are already defined in the Region enum for your SDK version.

If you are using version 1.2.34 or later of the SDK for Java, you can pass the new region name as a string using one of the following methods:

- To set the region on a previously created client:

```
client.setRegion("ca-toronto-1");
```

- To set a region when building a new client:

```
Identity identityClient = IdentityClient.builder()
        .region("ca-toronto-1")
        .build(provider);
```

If you are authenticating via instance principals, you can set the federationEndpoint with the following code. This method works with any version of the SDK.

```
InstancePrincipalsAuthenticationDetailsProvider provider =
InstancePrincipalsAuthenticationDetailsProvider.builder()
        .federationEndpoint("https://auth.ca-toronto-1.oraclecloud.com/v1/x509")
        .build();
```

OTHER REALMS

For regions in realms other than `oraclecloud.com`, you can use the following workarounds to reach new regions with earlier versions of the SDK.

To specify the endpoint:

```
AuthenticationDetailsProvider provider =
        new ConfigFileAuthenticationDetailsProvider(configurationFilePath, profile);

IdentityClient client = IdentityClient.builder()
        .endpoint("https://identity.ca-toronto-1.oraclecloud.com")
        .build(provider);
```

If you are authenticating via instance principals, you can set the endpoint and federationEndpoint via the following process:

```
InstancePrincipalsAuthenticationDetailsProvider provider =
InstancePrincipalsAuthenticationDetailsProvider.builder()
        .federationEndpoint("https://auth.ca-toronto-1.oraclecloud.com/v1/x509")
        .build();

IdentityClient identityClient = IdentityClient.builder()
        .endpoint("https://identity.ca-toronto-1.oraclecloud.com")
        .build(provider);
```

PAGINATED RESPONSES

Some APIs return paginated result sets, so you must check for additional items and if necessary, fetch the next page. You can do so manually or you can use an iterator.

MANUALLY FETCHING PAGES

The Response objects contain a method to fetch the next page token. If the token is null, there are no more items. If it is not null, you can make an additional request, by setting the token on the `Request` object, to get the next page of responses.



Note

Some APIs may return a token even if there are no additional results. Be sure to also check whether any items were returned and stop if there are none.

This example shows how to handle page tokens returned by the Object Storage API:

```
ObjectStorage client = ...;

ListBucketsRequest.Builder builder =
    ListBucketsRequest.builder().namespaceName(namespace);
String nextPageToken = null;
do {
    builder.page(nextPageToken);
    ListBucketsResponse listResponse = client.listBuckets(builder.build());
    List<Bucket> buckets = listResponse.getItems();
    // handle buckets
    nextPageToken = listResponse.getOpcNextPage();
} while (nextPageToken != null);
```

USING AN ITERATOR

Instead of manually working with page tokens, you can use an iterator. Each service client exposes a `getPaginators()` method that returns a `Paginator` object. This object contains methods to return objects of type Iterable. We support two approaches to using **iterable**:

- **Response Iterator**: You can iterate over the `Response` objects that are returned by the list operation. These are referred to as *ResponseIterators*, and their methods are suffixed with "ResponseIterator," for example, *listUsersResponseIterator*.

```
Iterable<ListUsersResponse> responseIterator = identityClient.getPaginators
().listUsersResponseIterator(request);
for (ListUsersResponse response : responseIterator) {
    for (User user : response.getItems()) {
        System.out.println(user);
    }
}
```

- **Record Iterator**: You can iterate over the resources/records that are listed. These are referred to as *RecordIterator*, and their methods are suffixed with "RecordIterator," for example, *listUsersRecordIterator*.

```
Iterable<User> recordIterator = identityClient.getPaginators().listUsersRecordIterator(request)
for (User user : recordIterator) {
    System.out.println(user);
}
```

EXCEPTION HANDLING

When handling an exception, you can get more information about the HTTP request that caused it, such as the status code or timeout. You can also get the request ID when handling a `BmcException` by using the `getOpcRequestId` method.

This example shows a try-catch block that handles a `BmcException`and prints the request ID.

```
ObjectStorage client = ...;
try {
    GetBucketResponse response = client.getBucket(
                GetBucketRequest.builder().namespaceName("myNamespace").bucketName("myBucket").build());
    String requestId = response.getOpcRequestId();
    System.out.println(requestId);
} catch (BmcException e) {
    String requestId = e.getOpcRequestId();
    System.out.println(requestId);
    e.printStackTrace();
}
```

Exceptions in the SDK for Java are runtime exceptions (unchecked), so they do not show up in method signatures. All APIs can throw a `BmcException`.

Examples

Examples of SDK usage can be found on GitHub, including:

- Example: Synchronous Object Storage
- Example: Asynchronous Object Storage
- Example: Create an instance
- Example: Get an instance's public IP address

The examples are also in the downloadable .zip file for the SDK. Examples for older versions of the SDK are in the downloadable .zip for the specific version, available on GitHub.

If you'd like to see another example not already covered, file a GitHub issue.

Running Examples

1. Download the SDK to a directory named `oci`. See GitHub for the download.
2. Unzip the SDK into the `oci` directory. For example: `tar -xf oci-java-sdk-dist.zip`
3. Create your configuration file in your home directory (`~/.oci/config`). See Configuring the SDK.
4. Use `javac` to compile one of the previous example classes from the `examples` directory, ex:

   ```
   javac -cp lib/oci-java-sdk-full-<version>.jar:third-party/lib/*
   examples/ObjectStorageSyncExample.java
   ```

5. You should now have a class file in the `examples` directory. Run the example:

   ```
   java -cp examples:lib/oci-java-sdk-full-<version>.jar:third-party/lib/* ObjectStorageSyncExample
   ```

Third-Party Dependencies and Shading

The SDK requires a number of third-party dependencies, which are available in the `third-party/lib` directory. To use the SDK library `lib/oci-java-sdk-full-<version>.jar`, all of the third-party dependencies in `third-party/lib` have to be on the class path.

The SDK also includes a second version of the SDK library, `shaded/lib/oci-java-sdk-full-shaded-<version>.jar`, which contains most of the third-party dependencies already. Only a few more third-party libraries in `shaded/third-party/lib` have to be on the class path when you use this version of the SDK library.

These two versions of the SDK library are functionally the same, however the second version, `shaded/lib/oci-java-sdk-full-shaded-<version>.jar` can simplify dealing with different versions of third-party dependencies. This is because all the dependencies that are included in `shaded/lib/oci-java-sdk-full-shaded-<version>.jar` were shaded, which means they will not interfere with other versions of themselves you may want to include along with this SDK.

You can use either `lib/oci-java-sdk-full-<version>.jar` or `shaded/lib/oci-java-sdk-full-shaded-<version>.jar`, but not both. When using `lib/oci-java-sdk-full-<version>.jar`, use all the third-party libraries in `third-party/lib`. When using `shaded/lib/oci-java-sdk-full-shaded-<version>.jar`, use all the third-party libraries in `shaded/third-party/lib`.

To use the shaded version of the SDK, replace the `javac` commands in steps 4 and 5 with the following:

- **Step 4:**

```
javac -cp shaded/lib/oci-java-sdk-full-shaded-<version>.jar:shaded/third-party/lib/*
examples/ObjectStorageSyncExample.java
```

- **Step 5:**

```
java -cp examples:shaded/lib/oci-java-sdk-full-shaded-<version>.jar:shaded/third-party/lib/*
ObjectStorageSyncExample
```

Troubleshooting

This section contains troubleshooting information for the Oracle Cloud Infrastructure SDK for Java.

OBJECTSTORAGE CLIENT DOES NOT CLOSE CONNECTIONS WHEN CLIENT IS CLOSED.

Too many file descriptors are opened up, and it takes too long to close existing ones. An exception may look like this:

```
Caused by: java.io.FileNotFoundException: classes/caspertest.pem (Too many open files)
at java.io.FileInputStream.open0(Native Method)
at java.io.FileInputStream.open(FileInputStream.java:195)
at java.io.FileInputStream.<init>(FileInputStream.java:138)
```

Use one of the following workarounds to fix this issue.

- Make this call before creating a client: `System.setProperty("http.keepAlive", "false");`
- Use this command line argument when running Java: `-Dhttp.keepAlive=false`

SERIALIZATION ERRORS WHEN MAKING REQUESTS OR HANDLING RESPONSES

If you encounter an `UnrecognizedPropertyException` error when handling a response to a call against the SDK for Java, this indicates that the version of the Jackson library in use does not support a feature that was injected at runtime from another dependency in your application's class path. This happens even if the `FAIL_ON_UNKNOWN_PROPERTIES` deserialization property is set to `false` for the configured `ObjectMapper`.

Solution:

Determine which version of Jackson libraries are referenced in your application’s class path and, if necessary, upgrade to version 2.9.5. For a complete list of Jackson libraries that the SDK for Java depends on, please refer to the pom.xml file that is hosted on GitHub.



Note

If you customize a client when instantiated in your application, ensure that you reference the preconfigured `ObjectMapper`from the `RestClientFactory` using the `RestClientFactory#getObjectMapper()` method.

An alternative solution is to to use the shaded version of the SDK for Java jar file, which includes a bundled version of the Jackson libraries.

ENCRYPTION KEY SIZE ERRORS

By default, the SDK for Java can only handle keys of 128 bit or lower key length. Users get "Invalid Key Exception" and "Illegal key size" errors when they use longer keys, such as AES256.

Use one of the following workarounds to fix this issue.

- Use a 128 bit key, such as AES128.
- Install the Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction from the following location: http://www.oracle.com/technetwork/java/javase/downloads/jce8-download-2133166.html

TROUBLESHOOTING SERVICE ERRORS

Any operation resulting in a service error will cause an exception of type com.oracle.bmc.model.BmcException to be thrown by the SDK. For information about common service errors returned by OCI, see API Errors.

Python SDK

The Python SDK enables you to write code to manage Oracle Cloud Infrastructure resources.

This SDK and sample is dual-licensed under the Universal Permissive License 1.0 and the Apache License 2.0; third-party content is separately licensed as described in the code.

Download: The Python SDK is available on GitHub or the Python Package Index (PyPi).

Documentation: Available on readthedocs.io.

Services Supported

- Announcements
- Audit
- Budgets
- Container Engine for Kubernetes
- Compute Autoscaling
- Core Services (Networking, Compute, Block Volume)
- Database
- DNS
- Email Delivery
- File Storage
- Health Checks
- IAM
- Key Management
- Load Balancing
- Monitoring
- Notifications
- Object Storage
- Resource Manager
- Search

- Streaming
- Web Application Acceleration and Security

Contact Us

CONTRIBUTIONS

Got a fix for a bug or a new feature you'd like to contribute? The SDK is open source and accepting pull requests on GitHub.

NOTIFICATIONS

To be notified when a new version of the Python SDK is released, subscribe to the Atom feed.

QUESTIONS OR FEEDBACK

- GitHub Issues: To file bugs and feature requests only
- Stack Overflow: Please use the oracle-cloud-infrastructure and oci-python-sdk tags in your post
- Developer Tools section of the Oracle Cloud forums
- My Oracle Support

Ruby SDK

The Ruby SDK enables you to write code to manage Oracle Cloud Infrastructure resources.

This SDK and sample are dual-licensed under the Universal Permissive License 1.0 and the Apache License 2.0; third-party content is separately licensed as described in the code.

Download: The Ruby SDK is available on GitHub or RubyGems.

Documentation: Ruby SDK documentation.

Services Supported

- Announcements
- Audit
- Budgets
- Container Engine for Kubernetes
- Compute Autoscaling
- Core Services (Networking, Compute, Block Volume)
- Database
- DNS
- Email Delivery
- File Storage
- Health Checks
- IAM
- Key Management
- Load Balancing
- Monitoring
- Notifications
- Object Storage
- Resource Manager
- Search
- Streaming
- Web Application Acceleration and Security

Contact Us

CONTRIBUTIONS

Got a fix for a bug or a new feature you'd like to contribute? The SDK is open source and accepting pull requests on GitHub.

NOTIFICATIONS

To be notified when a new version of the Ruby SDK is released, subscribe to the Atom feed.

QUESTIONS OR FEEDBACK

- GitHub Issues: To file bugs and feature requests only
- Stack Overflow: Please use the oracle-cloud-infrastructure and oci-ruby-sdk tags in your post
- Developer Tools section of the Oracle Cloud forums
- My Oracle Support

Go SDK

The Go SDK enables you to write code to manage Oracle Cloud Infrastructure resources.

This SDK and sample is dual-licensed under the Universal Permissive License 1.0 and the Apache License 2.0; third-party content is separately licensed as described in the code.

Download: Download the SDK from GitHub.

Documentation: Available on godoc.

Services Supported

- Announcements
- Audit
- Budgets

- Container Engine for Kubernetes
- Compute Autoscaling
- Core Services (Networking, Compute, Block Volume)
- Database
- DNS
- Email Delivery
- File Storage
- Health Checks
- IAM
- Key Management
- Load Balancing
- Monitoring
- Notifications
- Object Storage
- Resource Manager
- Search
- Streaming
- Web Application Acceleration and Security

Contact Us

CONTRIBUTIONS

Got a fix for a bug or a new feature you'd like to contribute? The SDK is open source and accepting pull requests on GitHub.

NOTIFICATIONS

To be notified when a new version of the Go SDK is released, subscribe to the Atom feed.

QUESTIONS OR FEEDBACK

- GitHub Issues: To file bugs and feature requests only
- Stack Overflow: Please use the oracle-cloud-infrastructure and oci-go-sdk tags in your post
- Developer Tools section of the Oracle Cloud forums
- My Oracle Support

Command Line Interface (CLI)

The CLI is a small footprint tool that you can use on its own or with the Console to complete Oracle Cloud Infrastructure tasks. The CLI provides the same core functionality as the Console, plus additional commands. Some of these, such as the ability to run scripts, extend the Console's functionality.

This CLI and sample is dual-licensed under the Universal Permissive License 1.0 and the Apache License 2.0; third-party content is separately licensed as described in the code.

The CLI is built on Python (version 2.7.5 or 3.5 or later), running on Mac, Windows, or Linux. The Python code makes calls to Oracle Cloud Infrastructure APIs to provide the functionality implemented for the various services. These are REST APIs that use HTTPS requests and responses. For more information, see About the API.

Installation: See Quickstart.

Reference: For help with a specific command, you can enter `help <command>` on the command line or view the Command Line Reference. This reference is derived from the APIs and help text in the Python source code.

Requirements

To install and use the CLI, you must have:

- An Oracle Cloud Infrastructure account
- A user created in that account, in a group with a policy that grants the desired permissions. This account user can be you, another person, or a system that calls the

API. For an example of how to set up a new user, group, compartment, and policy, see Adding Users. For a list of other typical Oracle Cloud Infrastructure policies, see Common Policies.

- A keypair used for signing API requests, with the public key uploaded to Oracle. Only the user calling the API should possess the private key. See Configuring the CLI.



Note

To use the CLI without a keypair, you can use token-based authentication. For more information, see Token-based Authentication for the CLI.

- Python version 2.7.5 or 3.5 or later, running on Mac, Windows, or Linux. Note that if you use the CLI Installer and do not have Python on your machine, the Installer offers to automatically install Python for you. If you already have Python installed on your machine, you can use the `python --version` command to find out which version is installed.
- If you require FIPS-compliance, see Using FIPS-validated Libraries.

Services Supported

- Announcements
- Audit
- Budgets
- Container Engine for Kubernetes
- Compute Autoscaling
- Core Services (Networking, Compute, Block Volume)
- Database
- DNS
- Email Delivery

- File Storage
- Health Checks
- IAM
- Key Management
- Load Balancing
- Monitoring
- Notifications
- Object Storage
- Resource Manager
- Search
- Streaming
- Web Application Acceleration and Security

Contact Us

CONTRIBUTIONS

Got a fix for a bug or a new feature you'd like to contribute? The SDK is open source and accepting pull requests on GitHub.

NOTIFICATIONS

To be notified when a new version of the CLI is released, subscribe to the Atom feed.

QUESTIONS OR FEEDBACK

- GitHub Issues: To file bugs and feature requests only
- Developer Tools section of the Oracle Cloud forums
- My Oracle Support

Quickstart

Using the installer script and the setup command is the fastest way to get up and running with the CLI.



Warning

Oracle recommends that you avoid using string values that include confidential information.

INSTALLING THE CLI

The installer script automatically installs the CLI and its dependencies, Python and virtualenv. Before running the installer, be sure you meet the Requirements.

MACOS, LINUX, AND UNIX

1. Open a terminal.
2. To run the installer script, run the following command.

```
bash -c "$(curl -L https://raw.githubusercontent.com/oracle/oci-
cli/master/scripts/install/install.sh)"
```

3. Respond to the Installation Script Prompts.

WINDOWS

1. Open the PowerShell console using the **Run as Administrator** option.
2. The installer enables auto-complete by installing and running a script. To allow this script to run, you must enable the RemoteSigned execution policy.
   To configure the remote execution policy for PowerShell, run the following command.

```
Set-ExecutionPolicy RemoteSigned
```

3. To run the installer script, run the following command.

```
powershell -NoProfile -ExecutionPolicy Bypass -Command "iex ((New-Object
System.Net.WebClient).DownloadString('https://raw.githubusercontent.com/oracle/oci-
cli/master/scripts/install/install.ps1'))"
```

4. Respond to the Installation Script Prompts.

INSTALLATION SCRIPT PROMPTS

The installation script prompts you for the following information.

- If you do not have a compatible version of Python installed:
  - Windows and Linux: You are prompted to provide a location for installing the binaries and executables. The script will install Python for you.
  - MacOS: You are notified that your version of Python is incompatible. You must upgrade before you can proceed with the installation. The script will not install Python for you.
- When prompted to upgrade the CLI to the newest version, respond with **Y** to overwrite an existing installation.
- When prompted to update your PATH, respond with **Y** to be able to invoke the CLI without providing the full path to the executable. This will add oci.exe to your PATH.

SETTING UP THE CONFIG FILE

Before using the CLI, you must create a config file that contains the required credentials for working with Oracle Cloud Infrastructure. You can create this file using a setup dialog or manually using a text editor.

USE THE SETUP DIALOG

To have the CLI walk you through the first-time setup process, use the `oci setup config` command. The command prompts you for the information required for the config file and the API public/private keys. The setup dialog generates an API key pair and creates the config file.

For more information about how to find the required information, see:

- Where to Get the Tenancy's OCID and User's OCID
- Regions and Availability Domains

MANUAL SETUP

If you want to set up the API public/private keys yourself and write your own config file, see SDK and Tool Configuration.



Tip

Use the `oci setup keys` command to generate a key pair to include in the config file.

NEXT STEPS

- For details on starting a session, see Starting a CLI Session
- Getting Started with the Command Line Interface provides an end-to-end walk-through of using the CLI to launch an instance.

Manual Installation

Instead of using the installer script as described in the Quickstart, you can manually install the CLI and its dependencies. Before proceeding, be sure you meet the Requirements.



Warning

Oracle recommends that you avoid using string values that include confidential information.

STEP 1: INSTALLING PYTHON

Python installation instructions vary for each operating system.

Windows and Windows Server 2008 R2

WINDOWS

Install Python from the Python Windows downloads page. During installation, choose to add Python to the PATH and/or environment variables (depending on the prompt).

WINDOWS SERVER 2008 R2

To install Python on this version of Windows Server you must install Python 2.7, or install Windows 2008 R2 Service Pack 1 (SP1) to the instance before installing later versions of Python.

Oracle Linux 7.3 and Oracle Linux 6

ORACLE LINUX

Some versions of Oracle Linux come with incompatible versions of Python, and may require additional components to install the CLI.

ORACLE LINUX 7.3

Before you install the CLI, run the following command on a new Oracle Linux 7.3 image.

```
sudo yum install gcc libffi-devel python-devel openssl-devel
sudo easy_install pip
```

ORACLE LINUX 6

On Oracle Linux 6, a newer version of Python is usually required. You can install a newer version alongside the existing version by downloading from Python, and then install the CLI in a virtual environment that uses the new version. To install the new version of Python, run the following commands.

```
sudo yum install gcc libffi-devel python-devel openssl-devel
sudo easy_install pip
curl -O https://www.python.org/ftp/python/3.6.0/Python-3.6.0.tgz
tar -xvzf Python-3.6.0.tgz
cd Python-3.6.0
```

```
./configure
make
sudo make install
```

CentOS 6 and CentOS 7

CentOS 6 and CentOS 7

Before you install the CLI, run the following commands on a new CentOS image.

```
sudo yum install gcc libffi-devel python-devel openssl-devel
sudo easy_install pip
```

Ubuntu 14.04 and Ubuntu 16.04

Ubuntu 14.04 and Ubuntu 16.04

Python should come installed. However, if you need to install Python, use the OS's normal package manager.

To install the additional required components, run the following commands.

```
sudo apt-get update
sudo apt-get install build-essential libssl-dev libffi-dev python-dev
sudo apt-get install python3-pippip3 install --upgrade pip
```

Step 2: Installing and Configuring virtualenv

`virtualenv` is a virtual environment builder that lets you create isolated Python environments. For Linux users, `virtualenv` is usually in a separate package from the main Python package.

Download the software and documentation from:

- Software: GitHub or PyPI.
- Documentation: Docs Virtualenv

Installing and Configuring virtualenv

After Python is installed, install and configure `virtualenv`.

1. To set up the environment for Python 2, use the following command.

   ```
   pip install virtualenv
   ```

2. To set up the environment for Python 3, use the following command.

   ```
   pip3 install virtualenv
   ```

   `virtualenv` typically installs to your Python directory. For example:

   ```
   /usr/local/share/python3/virtualenv
   ```

3. (Optional) To create a directory for storing your virtual environments, run the following command.

   ```
   mkdir -p myvirtualspaces/virtualenvs
   ```

4. To create a new virtual environment without any packages, run the following command.

   ```
   virtualenv myvirtualspaces/virtualenvs/cli-testing --no-site-packages
   ```

If you're installing a newer version of Python to run alongside an existing version, you can create a virtual environment that uses the new version.

To reference the new version of Python, run the following command with the `-p` parameter.

```
virtualenv -p /usr/local/bin/python3.6 cli-testing
```

STEP 3: INSTALLING THE COMMAND LINE INTERFACE

You can download the CLI from GitHub or install the package from Python Package Index (PyPI).

To install using the GitHub download:

- Download and unzip **oci-cli.zip**.
- Run the following command.

  ```
  pip install oci_cli-*-py2.py3-none-any.whl
  ```

To install using PyPI, run the following command.

```
pip install oci-cli
```

For information on how to start a CLI session, see Starting a CLI Session.

Installing Without a Virtual Environment

We do not recommend installing the CLI in your system-wide Python and suggest that instead you install the CLI using the installer or virtual environment.

In cases where you are trying to install the CLI in your system-wide Python using the latest pip version, you might encounter conflicts with some `distutils` installed packages. Following is an example error message when this occurs:

```
sudo pip install oci-cli
...
...
Cannot uninstall 'requests'. It is a distutils installed project and thus we cannot accurately determine
which files belong to it which would lead to only a partial uninstall.
```

Another option is to install the CLI for the user using the following command, although this approach is not supported:

```
pip install --user oci-cli
```

Token-based Authentication for the CLI

Token-based authentication for the CLI allows customers to authenticate their session interactively, then use the CLI for a single session without an API signing key. This enables customers using an identity provider that is not SCIM-supported to use a federated user account with the CLI and SDKs.

Requirements

The requirements are the same as those listed for the CLI in Requirements, except that instead of a SSH keypair, you need a web browser for the authentication process.

Starting a Token-based CLI Session

To use token-based authentication for the CLI on a computer with a web browser:

1. In the CLI, run the following command. This will launch a web browser.

   ```
   oci session authenticate
   ```

2. In the browser, enter your user credentials. This authentication information is saved to the `.config` file.

Validating a Token

To verify that a token is valid, run the following command:

```
oci session validate --config-file <path_to_config_file> --profile <profile_name> --auth security_token
```

You should receive a message showing the expiration date for the session. If you receive an error, check your profile settings.

Refreshing a Token

The default token TTL is set to 1 hour before it expires and can be refreshed within the validity period up to 24 hours.

To refresh the token, run the following command:

```
oci session refresh --profile <profile_name>
```

Starting a Token-based CLI Session without a Browser

To use token-based authentication for the CLI on a computer without a web browser, you must export a session from a web-enabled computer, then import it to the computer without a web browser.

Exporting from Source Computer

On the source computer with the browser:

1. In the CLI, run the following command:

   ```
   oci session authenticate
   ```

2. Enter the user credentials you wish to use on the target computer.
3. To export a zip file, run the following command:

   ```
   oci session export --profile <profile_name> --output-file <output_filename>
   ```

To verify the export, see Validating a Token.

Importing to Target Computer

On the target computer without the browser, run the following command in the CLI,:

```
oci session import --session-archive <path_to_exported_zip>
```

You can test the import by running the following:

```
oci iam region list --config-file <path_to_config_file> --profile <profile_name> --auth security_token
```

It should return a list of regions. Successful execution of this command verifies that the token authentication is working as expected.

Running Scripts on a Computer without a Browser

After importing the authentication to the target computer, you can run the CLI and SDKs by using the following settings.

For CLI

To run scripts on the CLI, append the following suffix:

```
--config-file <path_to_config_file> --profile <profile_name> --auth security_token
```

For SDKs

To run SDKs on the target computer, you must read in the token file, then use it to initialize the `SecurityTokenSigner`.

After creating a token file as shown in Starting a Token-based CLI Session, use the following process.



Note

These code samples demonstrate how to accomplish this using the Python SDK. For other SDKs, follow the same process, but adjust the syntax accordingly.

1. Read the token file from the `security_token_file` parameter of the `.config` file.

```
config = oci.config.from_file(profile_name='TokenDemo')
token_file = config['security_token_file']
token = None
with open(token_file, 'r') as f:
     token = f.read()
```

2. Read the private key specified by the `.config` file.

```
private_key = oci.signer.load_private_key_from_file(config['key_file'])
```

3. Create the initial SDK client which targets the user-specified region.

```
signer = oci.auth.signers.SecurityTokenSigner(token, private_key)
client = oci.identity.IdentityClient({'region': region}, signer=signer)
```

4. Make the identity request.

```
result = client.list_region_subscriptions(config['tenancy'])
```

Configuration

You can use these optional configurations to extend CLI functionality. The CLI supports using a file for CLI-specific configurations. You can:

- Specify a default profile.
- Set default values for command options so you don't have to type them into the command line.
- Define aliases for commands. For example, using "ls" as an alias for `list`.
- Define aliases for options. For example, using "--ad" as an alias for `--availability-domain`.
- Define named queries that are passed to the `--query` option instead of typing a JMESPath expression on the command line.

The default location and file name is `~/.oci/oci_cli_rc`. You can also explicitly specify this file with the `--cli-rc-file` option or by with the legacy `--defaults-file` option. For example:

```
# Uses the file from ~/.oci/oci_cli_rc
oci os bucket list

# Uses a custom file
oci os bucket list --cli-rc-file path/to/my/cli/rc/file
```

To set up an oci-cli-rc file, run the following command.

```
oci setup oci-cli-rc --file path/to/target/file
```

This command creates the file you specify that includes examples of default command aliases, parameter aliases, and named queries.



Note

If you are using Windows, you should use backslash as the directory separator in pathnames, instead of the forward slash.

SPECIFYING A DEFAULT PROFILE

Specify a default profile in the `OCI_CLI_SETTINGS` section of the CLI configuration file. The next example shows how to specify a default profile named IAD. The CLI looks for a profile named IAD in your `~/.oci/config` file, or any other file that you specify using the `--config-file` option.

```
[OCI_CLI_SETTINGS]
default_profile=IAD
```

You can also specify a default value for the `--profile` option using the `OCI_CLI_PROFILE` environment variable.

If a default profile value has been specified in multiple locations, the order of precedence is:

1. The value specified in the `--profile` option.
2. The value specified in the `OCI_CLI_PROFILE` environment variable.
3. The value specified in the `default_profile` field in the `OCI_CLI_SETTINGS` section of the CLI configuration file.

SPECIFYING DEFAULT VALUES

The CLI supports using a default values file so that you don't have to keep typing them into the command line. For example, instead of typing in a `--compartment-id` on each launch instance command or having to keep specifying the `--namespace` when using Object Storage commands. You can put this information in a default values file.

Default values can be applied at different levels, from general to specific:

- Globally, across all the CLI commands.
- To a particular service, such as Compute or Object Storage.
- To a specific group, such as commands related to exporting images.
- To a specific command.

Default values are treated hierarchically, with specific values having a higher order of precedence than general values. For example, if there is a globally defined value for `compartment-id` and a specific `compartment-id` defined for the `compute instance launch` command, the CLI uses the value for the `compute instance launch` instead of the global default.

DEFAULT VALUES FILE NAME AND LOCATION

When you start the CLI, the program looks for the default values file in `~/.oci/oci_cli_rc`. You can also specify a different file and location by using the `--cli-rc-file` option, as illustrated by the following:

```
# Uses the default values file from ~/.oci/oci_cli_rc
oci os bucket list

# Uses a custom default values file
oci os bucket list --cli-rc-file path/to/my/cli/custom-oci-cli-rc-file
```

COMMAND VALUE PRIORITY

If a value is provided on the command line also exists in `--cli-rc-file`, the value from the command line has priority. For a command with options that take multiple values, the values are taken entirely from the command line or from `--cli-rc-file`. The 2 sources aren't merged.

DEFAULTS VALUE FILE SYNTAX

The `--cli-rc-file` file can be divided into different sections with one or more keys per section.

Sections

In the next example, the file has two sections, with a key in each section. To specify which section to use, you use the `--profile` option in the CLI.

```
[DEFAULT]
compartment-id = ocid1.compartment.oc1..aaaaaaaal5zx25nzpgeyqd3gzijdlg3ieqeyrggnx7il26astxxhqoljnhwa
[ANOTHER_SECTION]
compartment-id = ocid1.compartment.oc1..aaaaaaaal3gzieyqdrggnx7xil26astxxhqol2pgjjdlieqeyg35nz5znhwa
```

Keys

Keys are named after command line options, but do not use a leading double hyphen (--). For example, the key for `--image-id` is `image-id`. You can specify keys for single values, multiple values, and flags.

- Keys for Single Values. The next example shows how to specify key values at different levels, and with different scope.

```
[DEFAULT]
# Defines a global default for bucket-name
bucket-name = my-global-default-bucket-name

# Defines a default for bucket-name, which applies to all 'compute' commands
compute.bucket-name = bucket-name-for-image-import-export

# Defines a default for bucket-name, which applies to all 'os object' commands (e.g., os object
get)
os.object.bucket-name = bucket-name-for-object-commands

# Defines a default for bucket-name, for the 'os object multipart list' command
os.object.multipart.list.bucket-name = bucket-name-for-multipart-list
```

- Keys for Multiple Values. Some options, such as `--include` and `--exclude` on the `oci os object bulk-upload` command can be specified more than once. For example:

  ```
  oci os object bulk-upload -ns my-namespace -bn my-bucket --src-dir my-directory --include *.txt --include *.png
  ```

  The next example shows how you would enter the `--include` values in the `--cli-rc-file` file

  ```
  [DEFAULT]
  os.object.bulk-upload.include =
                          *.txt
                          *.png
  ```

  In the previous example, one value is given for each line and each line must be indented underneath its key. You can use tabs or spaces and the amount of indentation doesn't matter. You can also put a value on the same line as the key, add more values on the following lines, and use a path statement for a value. For example:

  ```
  [DEFAULT]
  os.object.bulk-upload.include = *.pdf
                          *.txt
                          *.png
                          my-subfolder/*.tiff
  ```

- Keys for Flags. Some command options are flags, like `--force`, which uses a Boolean value. To set a flag for the `--force` option, use the following command.

  ```
  os.object.delete.force=true
  ```

SPECIFYING COMMAND ALIASES

Specify named queries in the OCI_CLI_COMMAND_ALIASES section of the CLI configuration file. There are two types of aliases, global aliases and command sequence aliases. The following example shows each type of alias.

```
[OCI_CLI_COMMAND_ALIASES]
# This is a global alias that lets you use "ls" instead of "list" for any list command in the CLI.
#
ls = list

# Command examples:
# oci os object ls or oci os compute ls
```

```
# This is a command sequence alias that lets you use "oci os object rm" instead of "oci os
# object delete".
# <alias> = <dot-separated sequence of groups and sub-groups>.<command or group to alias>
#
rm = os.object.delete

# Command example:
# <alias> = rm, <sequence of groups and sub-groups> = os object, <command or group to alias> = delete
```

If you want to define default values for options in your CLI configuration file, you can use the alias names you have defined. For example, if you have `-ls` as an alias for `--list`, you can define a default for an availability domain when listing instances by using the following command.

```
[DEFAULT]
compute.instance.ls.compartment-
id=ocid1.compartment.oc1..aaaaaaaal5zx25nzpgeyqd3gzijdlg3ieqeyrggnx7il26astxxhqoljnhwa
```

SPECIFYING OPTION ALIASES

Specify option aliases in the OCI_CLI_PARAM_ALIASES section of the CLI configuration file. Option aliases are applied globally. The following example shows some aliases for command options.

```
[OCI_CLI_PARAM_ALIASES]
# Option aliases either start with a double hyphen (--) or are a single hyphen (-) followed by a #
single letter. For example: --example-alias, -e
#
--ad = --availability-domain
--dn = --display-name
--egress-rules = --egress-security-rules
--ingress-rules = --ingress-security-rules
```

If you want to define default values for options in your CLI configuration file, you can use the alias names you have defined. For example, if you have `-ad` as an alias for `--availability-domain`, you can define a default for an availability domain when listing instances by using the following command.

```
[DEFAULT]
compute.instance.list.ad=xyx:PHX-AD-1
```

SPECIFYING NAMED QUERIES

If you use the `--query` parameter to filter or manipulate output, you can define named queries instead of using a JMESPath expression on the command line.

Specify named queries in the OCI_CLI_CANNED_QUERIES section of the CLI configuration file.

Examples of Named Queries

```
[OCI_CLI_CANNED_QUERIES]
#  For list results, this gets the ID and display-name of each item in the list.
#  Note that when the names of attributes have dashes in them they need to be surrounded
#  with double quotes. This query knows to look for a list because of the [*] syntax

get_id_and_display_name_from_list=data[*].{id: id, "display-name": "display-name"}

get_id_and_display_name_from_single_result=data.{id: id, "display-name": "display-name"}

#  Retrieves a comma separated string, for example:
#  ocid1.instance.oc1.phx.xyz....,cli_test_instance_675195,RUNNING
#
get_id_display_name_and_lifecycle_state_from_single_result_as_csv=data.[id, "display-name", "lifecycle-
state"] | join(`,`, @)

#  Retrieves comma separated strings from a list of results
#
get_id_display_name_and_lifecycle_state_from_list_as_csv=data[*].[join(`,`, [id, "display-name",
"lifecycle-state"])][]

#  Filters where the display name contains some text
#
filter_by_display_name_contains_text=data[?contains("display-name", `your_text_here`)]

#  Filters where the display name contains some text and pull out certain attributes(id and time-
created)
#
filter_by_display_name_contains_text_and_get_attributes=data[?contains("display-name", `your_text_
here`)].{id: id, timeCreated: "time-created"}

#  Get the top 5 results from a list operation
#
```

```
get_top_5_results=data[:5]

#  Get the last 2 results from a list operation
#
get_last_2_results=data[-2:]
```

You can reference any of these queries using this syntax: `query://<query name>`.

For example, to get id and display name from a list, run the following command.

```
oci compute instance list -c $C --query query://get_id_and_display_name_from_list
```

ENABLING AUTO-COMPLETE

If you used the CLI installer, you don't have to configure auto-complete because it's enabled automatically.

To enable auto-complete (tab completion) for a manual CLI installation, run the following command.

```
oci setup autocomplete
```

To enable auto-complete on a session by session basis, run the following command.

```
eval "$(_OCI_COMPLETE=source oci)"
```



Note

Support for Auto-complete on Windows

Auto-complete on Windows is only supported if you're using PowerShell. A script runs to enable this feature. However, you must change the PowerShell execution policy to RemoteSigned. To configure this policy, run the following command at the PowerShell command line.

```
Set-ExecutionPolicy RemoteSigned
```

USING FIPS-VALIDATED LIBRARIES

The CLI can be configured to use FIPS-validated libraries on Linux. The CLI is built on the Python SDK and leverages operating system level cryptographic libraries.

CONFIGURING THE ENVIRONMENT

1. Verify the installed version of OpenSSL is FIPS-compliant. Run the following command:

   ```
   openssl version
   ```

   If "fips" is not part of the version name, you should upgrade OpenSSL to a FIPS-compliant version. You can download the latest versions of OpenSSL at: https://www.openssl.org/source/

2. Determine the location of the FIPS-compliant version of libcrypto:

   ```
   ls -l /usr/lib64/libcrypto*
   ```

3. Set the environment variable OCI_CLI_FIPS_LIBCRYPTO_FILE to the location of libcrypto:

   ```
   export OCI_CLI_FIPS_LIBCRYPTO_FILE=</path/to/libcrypto.x.x.x>
   ```

   If you do not want to run this command at the start of every session, you can add it to your.bashrc or .bash_profile file.

   You can confirm that the environment variable is set properly with this command:

   ```
   set | grep OCI_CLI_FIPS_LIBCRYPTO_FILE
   ```

You can now proceed to the standard installation process outlined in Quickstart

VERIFYING THE CONFIGURATION

To verify that the CLI is using the library that you specified during Configuring the Environment, execute the following commands in Python. Be sure to do so in the same environment that the CLI uses.

```
import ssl
ssl.FIPS_mode()
```

This should return 1, indicating that SSL is using the library specified by the OCI_CLI_FIPS_LIBCRYPTO_FILE environment variable.

Using the CLI

This topic describes how to use the CLI to access Oracle Cloud Infrastructure and carry out service-related tasks. This topic assumes that you have configured the CLI and are ready to start using it.



Tip

Getting Started with the Command Line Interface provides an end-to-end walk-through of using the CLI to launch an instance.

STARTING A CLI SESSION

MACOS, LINUX, AND UNIX

To start a CLI session, run the following commands.

1. Open a terminal.
2. Change the working directory.

```
cd myvirtualspaces/virtualenvs/cli-testing/bin
```

3. Run the activate batch file.

```
source activate
```

To stop using the CLI, run the following command in a terminal.

```
deactivate
```

WINDOWS

To start a CLI session, run the following commands.

1. Open the Command Prompt using the **Run as administrator** option.
2. Change the working directory.

   ```
   cd myvirtualspaces/virtualenvs/cli-testing/Scripts
   ```

3. Run the activate batch file.

   ```
   activate
   ```

To stop using the CLI, run the following command from the command line.

```
deactivate
```



Warning

Avoid entering confidential information when providing resource names, descriptions, or other values that may expose sensitive information.

COMMAND LINE SYNTAX

Most commands must specify a service, followed by a resource type and then an action. The basic command line syntax is:

```
oci <service> <type> <action> <options>
```

For example, this syntax is applied as follows:

- `compute` is the <service>
- `instance` is the resource <type>
- `launch` is the <action>, and
- the rest of the command string consists of <options>.

The following command to launch an instance shows a typical command line construct.

```
oci compute instance launch --availability-domain "EMIr:PHX-AD-1" -c
ocid1.compartment.oc1..aaaaaaaal3gzijdlieqeyg35nz5zxil26astxxhqol2pgeyqdrggnx7jnhwa --shape
"VM.Standard1.1"   --display-name "Instance 1 for sandbox" --image-id
```

```
ocid1.image.oc1.phx.aaaaaaaaqutj4qjxihpl4mboabsa27mrpusygv6gurp47kat5z7vljmq3puq --subnet-id
ocid1.subnet.oc1.phx.aaaaaaaaypsr25bzjmjyn6xwgkcrgxd3dbhiha6lodzus3gafscirbhj5bpa
```



Warning

In the previous example, you can provide a friendly name for the instance using the `--display-name` option. Avoid entering confidential information when providing resource names or descriptions.

BASIC EXAMPLES

This section provides examples of basic operations using the CLI.



Note

Using Environment Variables for OCIDs

Several of the CLI examples use environment variables for OCIDs, such as:

- $T for a tenancy OCID
- $C for a compartment OCID

For example:

```
T=ocid1.tenancy.oc1..aaaaaaaaba3pv6wm2ytdrwrx32uzr4h25vkcr4jqa
e5f15p2b2qstifsfdsq

C=ocid1.compartment.oc1..aaaaaaaarhifmvrvuqtye5q66rck6copzqck3
ukc5fldrwpp2jojdcypxfga
```

To get a namespace, run the following command.

```
oci os ns get
```

To list compartments, run the following command.

```
oci iam compartment list -c $T
```

To get a list of buckets, run the following command.

```
oci os bucket list -ns mynamespace --compartment-id $C
```

To list users and limit the output, run the following command.

```
oci iam user list --compartment-id $T --limit 5
```

To add a user to a group, run the following command.

```
oci iam group add-user --user-id
ocid1.user.oc1..aaabcaaaxkkhhtmghvqqq7rgvzwuj3drwmtlsgz6sbfo7y4uc5sprzli377q --group-id
ocid1.group.oc1..aaabcaaa66plootq6uuwwxhfdw2lsdqtgeb6l4pjsv5eeuenxrauujj35b7b
```

GETTING HELP WITH COMMANDS

You can get help for any command using `--help`, `-h`, or `-?`. For example:

```
oci --help
```

```
oci os bucket -h
```

```
oci os bucket create -?
```

VIEWING ALL THE CLI HELP

You can view the command line help.

DETERMINING THE INSTALLED VERSION OF THE CLI

To get the installed version of the CLI, run the following command.

```
oci --version
```

USING DATES AND TIMES IN CLI COMMANDS

The CLI supports the following accepted date formats.

- UTC with milliseconds

  ```
  Format: YYYY-MM-DDTHH:mm:ss.sssTZD, Example: 2017-09-15T20:30:00.123Z
  ```

- UTC without milliseconds

  ```
  Format: YYYY-MM-DDTHH:mm:ssTZD, Example: 2017-09-15T20:30:00Z
  ```

- UTC with minute precision

```
Format: YYYY-MM-DDTHH:mmTZD, Example: 2017-09-15T20:30Z
```

- Timezone with milliseconds

```
Format: YYYY-MM-DDTHH:mm:ss.sssTZD, Example: 2017-09-15T12:30:00.456-08:00
```

- Timezone without milliseconds

```
Format: YYYY-MM-DDTHH:mm:ssTZD, Example: 2017-09-15T12:30:00-08:00
```

- Timezone with offset with minute precision

```
Format: YYYY-MM-DDTHH:mmTZD, Example: 2017-09-15T12:35-08:00
```

- Date Only (This date will be taken as midnight UTC of that day)

```
Format: YYYY-MM-DD, Example: 2017-09-15
```

- Epoch seconds

```
Example: 1412195400
```



Note

In our datetime formats, the `T` can be replaced with a space. For example, both `"2017-09-15 20:30:00.123Z"` and `2017-09-15T20:30:00.123Z` are acceptable. (Note that if you do not include the `T`, you must wrap the value in quotes.) We also support time zones with and without the colon. Both `+10:00` and `+1000` are acceptable.

MANAGING CLI INPUT AND OUTPUT

The CLI provides several options for managing command input and output.

PASSING COMPLEX INPUT

Complex input, such as arrays and objects with more than one value, are passed in JSON format and can be provided as a string at the command line, as a file, or as a command line string and as a file.

MacOS, Linux, or Unix

The following command shows how to pass two values for the `--metadata` object.

```
oci os bucket create -ns mynamespace --name mybucket --metadata '{"key1":"value1","key2":"value2"}' --
compartment-id ocid1.compartment.oc1..aaaaaaaarhifmvrvuqtye5q66rck6copzqck3ukc5fldrwpp2jojdcypxfga
```

Windows

On Windows, to pass complex input to the CLI as a JSON string, you must enclose the entire block in double quotes. Inside the block, each double quote for the key and value strings must be escaped with a backslash (\) character.

The following command shows how to pass two values for the `--metadata` object on Windows.

```
oci os bucket create -ns mynamespace --name mybucket --metadata "
{\"key1\":\"value1\",\"key2\":\"value2\"}" --compartment-id
ocid1.compartment.oc1..aaaaaaaarhifmvrvuqtye5q66rck6copzqck3ukc5fldrwpp2jojdcypxfga
```



Note

JSON Errors

The error message "Parameter '<PARAMETER NAME>' must be in JSON format." indicates that the value you passed for the parameter with name "PARAMETER NAME" was not valid JSON. This error is typically a result of the JSON string not being escaped correctly.

For more information about using JSON strings, see Advanced JSON Options

FORMAT OUTPUT AS A TABLE

By default, all responses to a command are returned in JSON format. For example, the following response is returned when you issue the command to get a list of regions.

```
{
  "data": [
    {
      "key": "FRA",
      "name": "eu-frankfurt-1"
    },
    {
      "key": "IAD",
      "name": "us-ashburn-1"
    },
    {
      "key": "ICN",
      "name": "ap-seoul-1"
    },
    {
      "key": "PHX",
      "name": "us-phoenix-1"
    },
    {
      "key": "LHR",
      "name": "uk-london-1"
    },
    {
      "key": "NRT",
      "name": "ap-tokyo-1"
    },
    {
      "key": "YYZ",
      "name": "ca-toronto-1"
    }
  ]
}
```

In some cases, readability can become an issue, which is easily resolved by formatting a response as a table. To get a response to a command formatted as a table, run the following command.

```
oci iam region list --output table
```

The following list of regions is returned as a two column table.

```
+-----+----------------+
| key | name           |
+-----+----------------+
| FRA | eu-frankfurt-1 |
| IAD | us-ashburn-1   |
| ICN | ap-seoul-1     |
| PHX | us-phoenix-1   |
| NRT | ap-tokyo-1     |
| LHR | uk-london-1    |
| YYZ | ca-toronto-1   |
+-----+----------------+
```

FILTER OUTPUT

You can filter output using the JMESPath query option for JSON. Filtering is very useful when dealing with large amounts of output. For example, run the following command with the output table option to get a list of images.

```
oci compute image list -c
ocid1.compartment.oc1..aaaaaaaapxgklgmujxjzx2ypptfjrcieq7rrob2u2zbesh3wlafsgthhqtea --output table
```

The image information is returned in table format, but too much data is returned, which overflows the width of the terminal. In addition, you may not need all the information that's returned.

```
| base-image-id | compartment-id | create-image-allowed | display-name
         | id | lifecycle-state | operating-system | operating-system-version | time-created
        |
+---------------+----------------+----------------------+-------------------------------------------
----------+-------------------------------------------------------------------------------------+----------
-------+-----------------+-------------------------+---------------------------------+| None
| None           | True                 | Windows-Server-2012-R2-Standard-Edition-VM-2017.07.25-0 | ocid
1.image.oc1.phx.aaaaaaaab2xgy6bijtudhsgsbgns6zwfqnkdb2bp4l4qap7e4mehv6bv3qca | AVAILABLE       | Windows
         | Serv
er 2012 R2 Standard  | 2017-07-25T23:59:59.311000+00:00 |
| None          | None           | True                 | Windows-Server-2012-R2-Standard-Edition-VM-
2017.04.03-0 | ocid
1.image.oc1.phx.aaaaaaaa53cliasgvqmutflwqkafbro2y4ywjebci5szc4eus5byy2e2b7ua | AVAILABLE       | Windows
         | Serv
er 2012 R2 Standard  | 2017-04-03T19:42:22.938000+00:00 |
| None          | None           | True                 | Windows-Server-2012-R2-Standard-Edition-BM-
```

```
2017.07.25-0 | ocid
1.image.oc1.phx.aaaaaaaadcegaay43eux6uap55fhp6lqaqh37xgocscktwm2yr7ql4pcykxq | AVAILABLE       | Windows
         | Serv
er 2012 R2 Standard  | 2017-07-25T20:55:37.937000+00:00 |
| None          | None           | True                 | Windows-Server-2012-R2-Standard-Edition-BM-
2017.04.13-0 | ocid1.image.oc1.phx.aaaaaaaa7xgecq2kt7tikqfrmshu6gwukoc3lcnf2iqtwmjyarlprp6j6lna |
AVAILABLE       | Windows          | Serv
er 2012 R2 Standard  | 2017-04-13T17:36:50.840000+00:00 |
| None          | None           | True                 | Windows-Server-2008-R2-Standard-Edition-VM-
2017.08.03-0 | ocid
1.image.oc1.phx.aaaaaaaaejmyrf52wf2blf7jd7y2dcrjvg6dyulfyp7d3r3oarc5ayka5liq | AVAILABLE       | Windows
         | Serv
er 2008 R2 Standard  | 2017-07-27T18:19:06.976000+00:00 |
| None          | None           | True                 | Oracle-Linux-7.4-2017.09.29-0
         | ocid
1.image.oc1.phx.aaaaaaaa3g2xpzlbrrdknqcjtzv2tvxcofjc55vdcmpxdlbohmtt7encpana | AVAILABLE       | Oracle
Linux     | 7.4
| 2017-10-05T22:36:17.246000+00:00 |
| None          | None           | True                 | Oracle-Linux-7.4-2017.08.25-1
         | ocid
1.image.oc1.phx.aaaaaaaajan2cd2g65tphpaiegiz4lbs422rdc73okcu7dt2uya6p5szywsa | AVAILABLE       | Oracle
Linux     | 7.4
| 2017-09-11T23:12:18.644000+00:00 |
| None          | None           | True                 | Oracle-Linux-7.4-2017.08.25-0
         | ocid
1.image.oc1.phx.aaaaaaaabifl2bmaygtu4riw3vcuowl5cqwdzdqzwndqneoybcfcn2pgyc6a | AVAILABLE       | Oracle
Linux     | 7.4| 2017-08-25T01:21:37.176000+00:00 |
```

You can limit the amount of data returned by combining the `--query` option with `--output table` to get the information you want from a command.

To get filtered image information returned in a table format, run the following command.

```
oci compute image list -c
ocid1.compartment.oc1..aaaaaaaapxgklgmujxjzx2ypptfjrcieq7rrob2u2zbesh3wlafsgthhqtea --output table --
query "data [*].{ImageName:\"display-name\", OCID:id}"
```

The previous command returns the following image information, formatted as a two column table.

```
+-------------------------------------------------------------+-----------------------------------------
-------------------------------------+
| ImageName                                                   | OCID
```

```
                                    |
+-----------------------------------------------------------+-----------------------------------------
-------------------------------------+
| Windows-Server-2012-R2-Standard-Edition-VM-2017.07.25-0 |
ocid1.image.oc1.phx.aaaaaaaab2xgy6bijtudhsgsbgns6zwfqnkdb2bp4l4qap7e4mehv6bv3qca |
| Windows-Server-2012-R2-Standard-Edition-VM-2017.04.03-0 |
ocid1.image.oc1.phx.aaaaaaaa53cliasgvqmutflwqkafbro2y4ywjebci5szc4eus5byy2e2b7ua |
| Windows-Server-2012-R2-Standard-Edition-BM-2017.07.25-0 |
ocid1.image.oc1.phx.aaaaaaaadcegaay43eux6uap55fhp6lqaqh37xgocscktwm2yr7ql4pcykxq |
| Windows-Server-2012-R2-Standard-Edition-BM-2017.04.13-0 |
ocid1.image.oc1.phx.aaaaaaaa7xgecq2kt7tikqfrmshu6gwukoc3lcnf2iqtwmjyarlprp6j6lna |
| Windows-Server-2008-R2-Standard-Edition-VM-2017.08.03-0 |
ocid1.image.oc1.phx.aaaaaaaaejmyrf52wf2blf7jd7y2dcrjvg6dyulfyp7d3r3oarc5ayka5liq |
| Oracle-Linux-7.4-2017.09.29-0                           |
ocid1.image.oc1.phx.aaaaaaaa3g2xpzlbrrdknqcjtzv2tvxcofjc55vdcmpxdlbohmtt7encpana |
| Oracle-Linux-7.4-2017.08.25-1                           |
ocid1.image.oc1.phx.aaaaaaaajan2cd2g65tphpaiegiz4lbs422rdc73okcu7dt2uya6p5szywsa |
| Oracle-Linux-7.4-2017.08.25-0                           |
ocid1.image.oc1.phx.aaaaaaaabifl2bmaygtu4riw3vcuowl5cqwdzdqzwndqneoybcfcn2pgyc6a |
| Oracle-Linux-7.3-2017.07.17-1                           |
ocid1.image.oc1.phx.aaaaaaaa7jvfm572d4ehcgh3ijapvhrt52voel33ispumnygi3kl7mph55ha |
| Oracle-Linux-7.3-2017.07.17-0                           |
ocid1.image.oc1.phx.aaaaaaaa5yu6pw3riqtuhxzov7fdngi4tsteganmao54nq3pyxu3hxcuzmoa |
| Oracle-Linux-6.9-2017.09.29-0                           |
ocid1.image.oc1.phx.aaaaaaaa2d243dmn6mj53zieyap5bdvtq7xfmr5kg5xulrldbjzdavaaoj6a |
| Oracle-Linux-6.9-2017.08.25-0                           |
ocid1.image.oc1.phx.aaaaaaaavlwrtcgz2mx6c4q4qg4gwvibx6g7xqkowe3tbbwjnifybwmexpnq |
| Oracle-Linux-6.9-2017.07.17-0                           |
ocid1.image.oc1.phx.aaaaaaaa3s4v5eamndtyghbo4bj2mhobkwjwbz3eowyy5cebmrsoxvoopixa |
| CentOS-7-2017.09.14-0                                   |
ocid1.image.oc1.phx.aaaaaaaauqtvzhqplzuyesb5tctig6qrwoavpnfiwdkvuynu7z646z72ahcq |
| CentOS-7-2017.07.17-0                                   |
ocid1.image.oc1.phx.aaaaaaaahmts5c5nktcnqsu6ppom72d7dnvkmqsoaavpsiklamn7qd3a7szq |
| CentOS-7-2017.04.18-0                                   |
ocid1.image.oc1.phx.aaaaaaaaamx6ta37uxltor6n5lxfgd5lkb3lwmoqurlpn2x4dz5ockekiuea |
| CentOS-6.9-2017.09.14-0                                 |
ocid1.image.oc1.phx.aaaaaaaagedr7qsbpxjylieetj7dy2r4xoq6p65v3i6y4simkhgyww2ibzxq |
| CentOS-6.9-2017.07.17-0                                 |
ocid1.image.oc1.phx.aaaaaaaalm3mr4lpsnzjev2nzmmkhpiy7yxu3456qyg7r4nvjieslp4yngtq |
| CentOS-6.8-2017.06.13-0                                 |
ocid1.image.oc1.phx.aaaaaaaauk5k4km4epm7fxj5ifuylvnyjfklmukqcg25clayx3ucuqizjbia |
| Canonical-Ubuntu-16.04-2017.08.22-0                     |
```

```
ocid1.image.oc1.phx.aaaaaaaalzhdvphf77qgvqo2apmve7o4s4jo77rluaf456qdzrtwmkq2xhra |
| Canonical-Ubuntu-16.04-2017.06.28-0                       |
ocid1.image.oc1.phx.aaaaaaaak2idogwetkehtdvo7m673ojuucpfxhybd3ehun7izzgjqi4c4gga |
| Canonical-Ubuntu-16.04-2017.05.16-0                       |
ocid1.image.oc1.phx.aaaaaaaae3a3oedsmmwsqu4dsrzntekefgq7vosngn4r6u6n5mis7dwpxxpa |
| Canonical-Ubuntu-14.04-2017.08.22-0                       |
ocid1.image.oc1.phx.aaaaaaaa2wjumduuoq6rqprrsmgu53eeyzp47vjztn355tkvsr3m2p57woqq |
| Canonical-Ubuntu-14.04-2017.07.10-0                       |
ocid1.image.oc1.phx.aaaaaaaaelnit3ag2zy3u5664shbjqgl6c33g2i436wix6xb5tqcsa7klnoa |
+-----------------------------------------------------------+-------------------------------------------
-------------------------------------+
```

For more information about the JMESPath query language for JSON, see JMESPath.

ADVANCED JSON OPTIONS

You can get the correct JSON format for command options and commands.

- For a command option, use `--generate-param-json-input` and specify the command option that you want to get the JSON for. To generate the JSON for creating or updating a security rule, run the following command.

  ```
  oci network security-list create --generate-param-json-input ingress-security-rules
  ```

  ### Response from the Command

  ```
  [
    {
      "icmpOptions": {
        "code": 0,
        "type": 0
        },
      "isStateless": true,
      "protocol": "string",
      "source": "string",
      "tcpOptions": {
      "destinationPortRange": {
        "max": 0,
        "min": 0
        },
  ```

```
  "sourcePortRange": {
    "max": 0,
    "min": 0
    }
  },
  "udpOptions": {
  "destinationPortRange": {
    "max": 0,
    "min": 0
    },
  "sourcePortRange": {
    "max": 0,
    "min": 0
    }
  }
  },
  {
  "icmpOptions": {
    "code": 0,
    "type": 0
    },
  "isStateless": true,
  "protocol": "string",
  "source": "string",
  "tcpOptions": {
  "destinationPortRange": {
    "max": 0,
    "min": 0
    },
  "sourcePortRange": {
    "max": 0,
    "min": 0
    }
  },
  "udpOptions": {
  "destinationPortRange": {
    "max": 0,
    "min": 0
    },
```

```
    "sourcePortRange": {
      "max": 0,
      "min": 0
      }
    }
  }
]
```

- For an entire command, use `--generate-full-command-json-input`. To generate the JSON for launching an instance, run the following command.

```
oci compute instance launch --generate-full-command-json-input
```

Response from the Command

```
{
  "assignPublicIp": true,
  "availabilityDomain": "string",
  "compartmentId": "string",
  "displayName": "string",
  "extendedMetadata": {
  "string1": {
    "string1": "string",
    "string2": "string"
   },
  "string2": {
    "string1": "string",
    "string2": "string"
   }
  },
  "hostnameLabel": "string",
  "imageId": "string",
  "metadata": {
    "string1": "string",
    "string2": "string"
    },
  "privateIp": "string",
  "shape": "string",
  "skipSourceDestCheck": true,
  "subnetId": "string",
```

```
  "vnicDisplayName": "string"
}
```

ORDER OF PRECEDENCE FOR JSON INPUT

The CLI supports combining arguments on the command line with file input. However, if the same values are provided in a file and on the command line, the command line takes precedence.

USING A JSON FILE FOR COMPLEX INPUT

You can pass complex input from a file by referencing it from the command line. For Windows users, this removes the requirement of having to escape JSON text. You provide a path to the file using the `file://` prefix.

Path Types

Using `testfile.json` as an example, the following types of paths are supported.

- Relative paths from the same directory, for example: `file://testfile.json` and `file://relative/path/to/testfile.json`
- Absolute paths on Linux, MacOS or Unix, for example: `file:///absolute/path/to/testfile.json`
- Full file paths on Windows, for example: `file://C:\path\to\testfile.json`



Note

File Path Expansions

File path expansions, such as "~/", "./", and "../", are supported. On Windows, the "~/" expression expands to your user directory, which is stored in the %USERPROFILE% environment variable. Using environment variables in paths is also supported.

File Locations

The following file locations are supported.

- Your home directory.

```
oci os bucket create -ns mynamespace --name mybucket --compartment-id
ocid1.compartment.oc1..aaaaaaaarhifmvrvuqtye5q66rck6copzqck3ukc5fldrwpp2jojdcypxfga --metadata
file://~/testfile.json
```

- The current directory.

```
oci os bucket create -ns mynamespace --name mybucket --compartment-id
ocid1.compartment.oc1..aaaaaaaarhifmvrvuqtye5q66rck6copzqck3ukc5fldrwpp2jojdcypxfga --metadata
file://testfile.json
```

- The /tmp directory (Linux, Unix, or MacOS).

```
oci os bucket create -ns mynamespace --name mybucket --compartment-id
ocid1.compartment.oc1..aaaaaaaarhifmvrvuqtye5q66rck6copzqck3ukc5fldrwpp2jojdcypxfga --metadata
file:///tmp/testfile.json
```

- The C:\temp directory (Windows).

```
oci os bucket create -ns mynamespace --name mybucket --compartment-id
ocid1.compartment.oc1..aaaaaaaarhifmvrvuqtye5q66rck6copzqck3ukc5fldrwpp2jojdcypxfga --metadata
file://C:\temp\testfile.json
```

EXAMPLES OF USING A JSON FILE AS INPUT

The examples in this section use JSON that's generated for a command option and an entire command. The JSON is saved in a file, edited, and then used as command line input.

Use File Input for a Command Option

This end-to-end example shows how to generate the JSON for a security list id option used to create a subnet. The JSON is saved in a file, edited, and then used as command line input.

Use a JSON File as Input for a Security List Option

1. To generate the JSON for the `security-list-ids` option, run the following command.

```
oci network subnet create --generate-param-json-input security-list-ids
```

2. Create a file and add the following content, which was returned in step 1. This content doesn't have to be escaped or on a single line, it just has to contain valid JSON.

```
[
  "string",
  "string"
]
```

3. Edit the file and replace the "string" values with values, as shown in the following example.

```
[
  "ocid1.securitylist.oc1.phx.aaaaaaaaw7c62ybv4676muq5tdrwup3v2maiquhbkbh4sf75tjcf5dm6kvlq",
  "ocid1.securitylist.oc1.phx.aaaaaaaa7snx4jh5drwo2h33rwcdqev6elir55hnrhi2yfndjfons5rcqk4q"
]
```

4. Save the file as "security-list.json".
5. To create the subnet using "security-list.json" as input, run the following command.

```
oci network subnet create --vcn-id
ocid1.vcn.oc1.phx.aaaaaaaa6wmuahgxejkv7ukyruqdrwlmrumtl6vyisxxxavagiqw2eeet2sa -c
ocid1.compartment.oc1..aaaaaaaal3gzijdliedxxhqol2rggndrwyg35nz5zxil26astpgeyq7jnhwa  --
availability-domain "EMIr:PHX-AD-1" --display-name TESTSUB --dns-label "testinstances" --cidr-
block "10.0.0.0/16" --security-list-ids file://security-list.json
```

Use File Input for an Entire Command

This end-to-end example shows how to generate the JSON to create a virtual cloud network (VCN). The JSON is saved in a file, edited, and then used as command line input.

Use a JSON File as Input to Create a VCN

1. To generate the JSON needed to create a VCN, run the following command.

```
oci network vcn create --generate-full-command-json-input
```

2. Create a file and add the following content, which was returned in step 1. This content doesn't have to be escaped or on a single line, it just has to contain valid JSON.

```
{
  "cidrBlock": "string",
  "compartmentId": "string",
  "displayName": "string",
  "dnsLabel": "string"
}
```

3. Edit the file and replace the "string" values with values, as shown in the following example.

```
{
  "cidrBlock": "10.0.0.0/16",
  "compartmentId":
"ocid1.compartment.oc1..aaaaaaaal3gzijdliedxxhqol2rggndrwyg35nz5zxil26astpgeyq7jnhwa",
  "displayName": "TestVCN",
  "dnsLabel": "testdns"
}
```

4. Save the file and name it "create-vcn.json"
5. To create the VCN using "create-vcn.json" as input, run the following command.

```
oci network vcn create --from-json file://create-vcn.json
```

ADVANCED EXAMPLES

The following examples show how you can use the CLI to complete complex tasks in Oracle Cloud Infrastructure.

WORKING WITH OBJECT STORAGE

You can use the CLI for several object operations with the Object Storage service.

Uploading and Downloading Files

Objects can be uploaded from a file or from the command line (STDIN), and can be downloaded to a file or to the command line (STDOUT).

Upload an object:

```
oci os object put -ns mynamespace -bn mybucket --name myfile.txt --file /Users/me/myfile.txt --metadata
'{"key1":"value1","key2":"value2"}'
```

Upload object contents from the command line (STDIN):

```
oci os object put -ns mynamespace -bn mybucket --name myfile.txt --file <--'object content'
```

Download an object:

```
oci os object get -ns mynamespace -bn mybucket --name myfile.txt --file /Users/me/myfile.txt
```

Print object contents to the command line (STDOUT):

```
oci os object get -ns mynamespace -bn mybucket --name myfile.txt --file -
```

Bulk Operations in Object Storage

The CLI supports the following bulk operations in Object Storage:

- Uploading files in a directory and all its subdirectories to a bucket

  ```
  # Upload all the files in a directory.
  oci os object bulk-upload -ns mynamespace -bn mybucket --src-dir path/to/upload/directory
  ```

- Downloading all objects, or all the objects that match a specified prefix, in a bucket

  ```
  # Download all the objects.
  oci os object bulk-download -ns mynamespace -bn mybucket --download-dir
  path/to/download/directory

  # Download all the objects that match the specified prefix.
  oci os object bulk-download -ns mynamespace -bn mybucket --download-dir
  path/to/download/directory --prefix myprefix
  ```

- Deleting all objects, or all the objects that match a specified prefix, in a bucket

  ```
  # Delete all the objects.
  oci os object bulk-delete -ns mynamespace -bn mybucket

  # Delete objects that match the specified prefix.
  oci os object bulk-delete -ns mynamespace -bn mybucket --prefix myprefix
  ```

Bulk operations support several options that let you:

- Overwrite or skip files/objects using `--overwrite` or `--no-overwrite`. (**Note**: If you pass neither of these options you are prompted for confirmation every time there is

something to overwrite.)

- Limit delete, upload, or download operations using `--prefix` and/or `--delimiter`
- Preview a bulk deletion with `--dry-run`

To get more information about the commands for bulk operations, run the following help commands:

```
# bulk-upload
oci os object bulk-upload -h

# bulk-download
oci os object bulk-download -h

# bulk-delete
oci os object bulk-delete -h
```

Multipart Operations in Object Storage

Multipart operations for Object Storage include object uploads and downloads.

MULTIPART UPLOADS

Large files can be uploaded to Object Storage in multiple parts to speed up the upload. By default, files larger than 128 MiB are uploaded using multipart operations. You can override this default by using the `--no-multipart` option.

You can configure the following options for the `oci os object put` command:

- `--no-multipart` overrides an automatic multipart upload if the object is larger than 128 MiB. The object is uploaded as a single part, regardless of size.
- `--part-size` in MiB, to use in a multipart operation. The default part size is 128 MiB and a part size that you specify must be greater than 10 MiB. If the object is larger than the `--part-size`, it is uploaded in multiple parts.
- `--parallel-upload-count`, to specify the number of parallel operations to perform. You can use this value to balance resources and upload times. A higher value may improve times but consume more system resources and network bandwidth. The default value is 10.

The `--resume-put` command allows you to resume a large file upload in cases where the upload was interrupted.



Note

Multipart Uploads from STDIN

Objects uploaded from STDIN are uploaded in multiple parts. If the object content is smaller than 10 MiB, the upload is only 1 part, and the MultipartUpload API is used for the upload. Specifying `--no-multipart` when uploading from STDIN will result in an error.

The following example shows the command for a multipart upload if the object is larger than 200 MiB.

```
oci os object put -ns my-namespace -bn my-bucket --file path/to/large/file --part-size 200
```

For more information about multipart uploads, see Using Multipart Uploads.

MULTIPART DOWNLOADS

Large files can be downloaded from Object Storage in multiple parts to speed up the download.

You can configure the following options for the `oci os object get` command:

- `--multipart-download-threshold` lets you specify the size, in MiB at which an object should be downloaded in multiple parts. This size must be at least 128 MiB.
- `--part-size`, in MiB, to use for a download part. This gives you the flexibility to use more (smaller size) or fewer (larger size) parts as appropriate for your requirements. For example, compute power and network bandwidth. The default minimum part size is 120 MiB.
- `--parallel-download-count` lets you specify how many parts are downloaded at the same time. A higher value may improve times but consume more system resources and network bandwidth. The default value is 10.

The following example shows the command to download any object with a size greater than 500 MiB. The object is downloaded in 128 MiB parts.

```
oci os object get -ns my-namespace -bn my-bucket --name my-large-object --multipart-download-threshold
500 --part-size 128
```

Upgrading the CLI

If you installed the CLI manually, use one of the following commands to upgrade the CLI.

- To upgrade a standard installation, run the following command.

  ```
  pip install oci-cli --upgrade
  ```

- To upgrade a standard virtualenv installation, run the following command.

  ```
  cli-testing/bin/pip install oci-cli --upgrade
  ```

If you installed the CLI using the install script, use the following process to upgrade the CLI:

- Run the install script and specify the same install directory.
- When prompted, reply **Y** to overwrite the existing installation.

Uninstalling the CLI

FOR MANUAL INSTALLATIONS

If you manually installed the CLI using pip, run the following command:

```
pip uninstall oci-cli
```

If you manually installed the CLI in a virtualenv, run the following command:

```
<path/to/virtualenv>/bin/pip uninstall oci-cli
```

FOR SCRIPT INSTALLATIONS

If you used the install script and the default installation location, you should delete the following directories.

On Windows:

- %USERPROFILE%/lib
- %USERPROFILE%/bin

On Mac:

- $HOME/lib
- $HOME/bin

If you used the install script, but installed to a custom location, you should delete the directories at that location.

UNINSTALLING PYTHON

The script also installs Python as a dependency if it was not already installed. In Windows 10, you can uninstall Python in Control Panel or at the command line.

USING CONTROL PANEL

To uninstall Python in Control Panel, select **Programs and Features**. Right-click Python and select **Uninstall**.

For more information, see Repair or remove programs in Windows 10.

USING THE COMMAND LINE

To uninstall Python at the command line, run the following command:

```
msiexec /x python<version>.msi
```

If you do not have the MSI file, you can also use the package or product code. For more information, see Using the Windows Installer.

Troubleshooting the CLI

This topic describes how to resolve issues that you might encounter when installing Python or the CLI, or when using the CLI.

Service Errors

Any operation resulting in a service error causes an error of type "ServiceError" to be returned by the CLI. For information about common service errors that Oracle Cloud Infrastructure returns, see API Errors.

Oracle Linux Permissions Issues

On Oracle Linux 7.3, if you encounter permission issues when running `pip install`, you might need to use `sudo`.

oci Command Not Found

If the `oci` command isn't found, this can be caused by one of the following reasons:

- pip installed the package to a different virtual environment than your active one.
- You switched to a different active virtual environment after you installed the CLI.

To determine where the CLI is installed, run the `which pip` and `which oci` commands.

Wheel File Won't Install

If the wheel file won't install, verify that pip is up to date. To update pip, run the `pip install -U pip` command. Try to install the wheel again.

Windows Issues

If the `oci` command isn't found, make sure that the `oci.exe` location is in your path (for example, the `Scripts` directory in your Python installation).

Contact Information

If you want to contribute ideas, report a bug, get notified about updates, have questions, or want to give feedback, use one of the following links.

Contributions

Got a fix for a bug, or a new feature you'd like to contribute? The CLI is open source and accepting pull requests on GitHub.

NOTIFICATIONS

To be notified when a new version of the CLI is released, subscribe to the Atom feed.

QUESTIONS OR FEEDBACK

Ways to get in touch:

- GitHub: To file bugs and feature requests only.
- Stack Overflow: Use the oracle-cloud-infrastructure and oci-cli tags in your post.
- Developer Tools section of the Oracle Cloud forums
- My Oracle Support

Other Tools and Plug-ins

Tools and Plug-ins for SDKs and CLI

Oracle Cloud Infrastructure provides additional developer tools for automating processes and facilitating development.

Toolkit for Eclipse - The toolkit is an open source plug-in for the Eclipse Integrated Development Environment (IDE) that enables Java developers to code and deploy applications more quickly and efficiently.

- **Documentation**: Toolkit for Eclipse
- **Download**: To build and install the Toolkit, clone the GitHub repository then follow instructions in Getting Started with Toolkit for Eclipse.

HDFS Connector for Object Storage - Read and write data with your Apache Hadoop application to and from the Oracle Cloud Infrastructure Object Storage service. Building the connector relies on Maven artifacts that are provided by the SDK for Java.

- **Documentation**: HDFS Connector for Object Storage
- **Download**: GitHub

DevOps Tools and Plug-ins

Oracle Cloud Infrastructure provides a number of DevOps tools and plug-ins for working with Oracle Cloud Infrastructure services. These can simplify provisioning and managing infrastructure or enable automated testing and continuous delivery.

Terraform Provider - Manage "infrastructure as code" with this component that connects Terraform to a given Oracle Cloud Infrastructure service.

- **Documentation**: Terraform Provider
- **Download**: GitHub

Ansible Modules - Automate provisioning and configuring of Oracle Cloud Infrastructure resources, such as Compute, Load Balancing, and Database services.

- **Documentation**: Ansible Modules
- **Download**: GitHub

Chef Knife Plug-in - Manage Oracle Cloud Infrastructure resources with Chef Knife, a command line tool that provides an interface between a local chef-repo and the Chef server.

- **Documentation**: Chef Knife Plug-in
- **Download**: GitHub

Compute Jenkins Plug-in - Bring up and down services or nodes as required to serve Jenkins Build Jobs and dynamically allocate Oracle Cloud Infrastructure resources for continuous integration tasks.

- **Documentation**: Compute Jenkins Plug-in
- **Download**: GitHub

Grafana Plug-in - Visualize metrics from the Monitoring service in your Grafana instance.

- **Documentation**: Grafana Plug-in
- **Download**: GitHub

Terraform Kubernetes Installer - Provision and configure the resources needed to run a highly available and configurable Kubernetes cluster.

- **Download**: GitHub

Kubernetes Volume Provisioner - Enable dynamic provisioning of storage resources when running Kubernetes on Oracle Cloud Infrastructure.

- **Download**: GitHub

DevOps Integrations

- Jenkins X Integration: Create a new Kubernetes cluster on Oracle Cloud Infrastructure Container Engine for Kubernetes.
- Packer Integration: Create reusable custom images.

Other Services and Features for DevOps

Oracle Cloud Infrastructure provides other services and features relevant to DevOps professionals.

- Container Engine for Kubernetes (OKE)
  Reliably build, deploy, and manage cloud-native containerized applications. You specify the compute resources that your applications require, and Container Engine for Kubernetes provisions them on Oracle Cloud Infrastructure in an existing tenancy.
- Oracle Cloud Infrastructure Registry
  Store, share, and manage development artifacts like Docker images. As Oracle Cloud Infrastructure Registry is managed by Oracle, your applications are deployed reliably and you don't have to deal with operational issues.

Toolkit for Eclipse

The Oracle Cloud Infrastructure Toolkit for Eclipse is an open source plug-in for the Eclipse Integrated Development Environment (IDE). The toolkit provides a set of features that help developers connect to Oracle Cloud Infrastructure from within Eclipse. For example, by using

the toolkit, you can deploy a project to the cloud by using the Object Storage feature to upload multiple files in one click. The Compute feature enables you to start a compute instance or restart it if needed. You can also switch between multiple accounts and regions from the Eclipse IDE.

Download: To install the Toolkit, download the `com.oracle.oci.eclipse.zip` toolkit from the releases section on GitHub, then follow the instructions in Getting Started with Toolkit for Eclipse.

Requirements

To use the Oracle Cloud Infrastructure Toolkit for Eclipse, you must have the following:

- An Oracle Cloud Infrastructure account
- A user created in that account, in a group with a policy that grants the desired permissions. This can be a user for yourself, or another person/system that needs to call the API. For an example of how to set up a new user, group, compartment, and policy, see Adding Users. For a list of typical policies you may want to use, see Common Policies.
- A key pair used for signing API requests, with the public key uploaded to Oracle. For more information on generating and uploading keys, see Required Keys and OCIDs.
- Java 8
- Maven
- SDK for Java
- Eclipse IDE for Java Developers 4.3 or later

Services Supported

- Compute
- Object Storage

Contact Us

CONTRIBUTIONS

Got a fix for a bug or a new feature you'd like to contribute? The plug-in is open source and accepting pull requests on GitHub.

NOTIFICATIONS

To be notified when a new version of the toolkit is released, subscribe to the Atom feed.

QUESTIONS OR FEEDBACK

- GitHub Issues: To file bugs and feature requests only
- Developer Tools section of the Oracle Cloud forums
- My Oracle Support

Getting Started with Toolkit for Eclipse

DOWNLOADING THE TOOLKIT

You can download the `com.oracle.oci.eclipse.zip` toolkit from the releases section on GitHub.

INSTALLING THE TOOLKIT

After building the toolkit, launch the Eclipse IDE.

1. From the top navigation bar, select **Help > Install New Software...**
2. In Install dialog, click **Add...**
3. In the Add Repository dialog, click **Archive...**
4. In the right pane of the Repository Archive window, select the zip file containing the toolkit. Click **Open**.
5. In the Add Repository dialog click **Add**.
6. In the Available Software dialog, select **Oracle Cloud Infrastructure Toolkit for**

Eclipse, then click **Next**.

7. In the Install Details dialog, click **Finish**.

CONFIGURING THE TOOLKIT

ORACLE CLOUD INFRASTRUCTURE PREFERENCES

Before you can use the toolkit, you must configure the Oracle Cloud Infrastructure Preferences in the Eclipse IDE. This process will provide the necessary identifiers and credentials so the toolkit can connect to your Oracle Cloud Infrastructure account. For more information, see Required Keys and OCIDs.

1. From the top navigation bar, select **Preferences > Oracle Cloud Infrastructure Preferences** .
2. For **Profile Name**, provide a short descriptive name.
3. From the **Region** dropdown, select your region.
4. Enter your **User OCID** and **Tenancy OCID**. For information on how to locate this information, see Where to Get the Tenancy's OCID and User's OCID.
5. For **Key File**, click **Browse** and select the appropriate file. For more information, see How to Generate an API Signing Key.
6. Enter the **Fingerprint** for the Key File. For more information, see How to Get the Key's Fingerprint.
7. Enter the **Passphrase**, if you created one for the key pair. If not, leave this field blank.
8. Click **Save Profile**.
9. Click **Apply and Close**.

PROXY SETTINGS

If you are on a network that uses a proxy to connect to the internet, you must configure Eclipse proxy settings. For more information, see Network Connections in the Eclipse IDE Documentation.

UNINSTALLING THE TOOLKIT

Launch the Eclipse IDE.

1. From the top navigation bar, select **Help > About Eclipse IDE**
2. Click **Installation Details**.
3. In the Installation Details window, select the Installed Software tab.
4. Select Oracle Cloud Infrastructure Toolkit for Eclipse and click **Uninstall...**
5. In the Uninstall dialog, confirm the items to be uninstalled then click **Finish**.
6. Click **Restart**.

Using Toolkit for Eclipse

After configuring the Oracle Cloud Infrastructure Preferences, you can connect to your tenancy via Eclipse and use the Oracle Cloud Infrastructure Explorer to view and update your resources. You can also switch between different accounts saved in the profile.

To change the region, click the region icon in the Explorer navigation bar and select from the dropdown.

To change the compartment, click the compartment icon and select from the dropdown.

Using the Toolkit for Eclipse with Compute Instances

In the Oracle Cloud Infrastructure Explorer, double-click **Compute** to view available resources.

Double-click **Instances** to display a list of instances. You can right-click each one in the list to start, stop, or reboot it.

You can also double-click **Block Volumes** to view a list of block volumes.

Using the Toolkit for Eclipse with Object Storage

In the Oracle Cloud Infrastructure Explorer, double-click Object Storage to view available resources.

Working with Buckets

To create a bucket, right-click **Object Storage** and select **Create New Bucket**.

To view content and details, double-click or right-click the bucket and select **Open Bucket**.

To delete a bucket, view the bucket's details and click **Delete Bucket**.

WORKING WITH OBJECTS

To upload an object, select the destination bucket and view its details. Right-click the Object list and select **Upload Object**. You can also drag one or more files to the Object list to upload.

To download one or more objects, right-click and select **Download Object**.

To delete one or more objects, right-click and select **Delete Object**.

HDFS Connector for Object Storage

The Hadoop Distributed File System (HDFS) Connector lets your Apache Hadoop application read and write data to and from the Oracle Cloud Infrastructure Object Storage service.

This SDK and sample is dual-licensed under the Universal Permissive License 1.0 and the Apache License 2.0; third-party content is separately licensed as described in the code.

- **Services supported:** Object Storage
- **Download:** GitHub
- **API Documentation:** HDFS Connector API Reference

Requirements

To use the HDFS connector, you must have:

- An Oracle Cloud Infrastructure account.
- A user created in that account, in a group with a policy that grants the desired permissions for any bucket you want to use. This can be a user for yourself, or another person/system that needs to call the API. For an example of how to set up a new user, group, compartment, and policy, see Adding Users. For a basic Object Storage policy, see Let Object Storage admins manage buckets and objects.
- Java 8

- A TTL value of 60. For more information, see Configuring JVM TTL for DNS Name Lookups.

Credentials and Passwords

If you use an encrypted PEM file for credentials, the passphrase will be read from configuration using the `getPassword` Hadoop Configuration method. The `getPassword` option checks for a password in a registered security provider. If the security provider doesn't contain the requested key, it will fallback to reading the plaintext passphrase directly from the configuration file.

Configuring JVM TTL for DNS Name Lookups

The Java Virtual Machine (JVM) caches DNS responses from lookups for a set amount of time, called *time-to-live* (TTL). This ensures faster response time in code that requires frequent name resolution.

The JVM uses the networkaddress.cache.ttl property to specify the caching policy for DNS name lookups. The value is an integer that represents the number of seconds to cache the successful lookup. The default value for many JVMs, `-1`, indicates that the lookup should be cached forever.

Because resources in Oracle Cloud Infrastructure use DNS names that can change, we recommend that you change the the TTL value to 60 seconds. This ensures that the new IP address for the resource is returned on next DNS query. You can change this value globally or specifically for your application:

- To set TTL globally for all applications using the JVM, add the following in the `$JAVA_HOME/jre/lib/security/java.security` file:

  ```
  networkaddress.cache.ttl=60
  ```

- To set TTL only for your application, set the following in your application's initialization code:

  ```
  java.security.Security.setProperty("networkaddress.cache.ttl" , "60");
  ```

Installation

Copy the bundled jars from lib and third-party/lib the to each node of the Hadoop cluster so that they are included in Hadoop's CLASSPATH.

SDK for Java and Maven Artifacts

Building an HDFS connector relies on Maven artifacts that are provided by the SDK for Java. To obtain the artifacts, you must download the SDK for Java and build it locally. You can then build the HDFS connector.



Important

The SDK for Javafile version that you download from the Oracle Releases page must match the HDFS connector version, which you can find in the `hdfs-connector/pom.xml` file in the dependency tag block that has the `groupId` attribute.

Properties

You can set the following HDFS Connector properties in the `core-site.xml` file. The BmcProperties page lists additional properties that you can configure for a connection to Object Storage.

Property	Description	Type	Required
`fs.oci.client.hostname`	URL of the host endpoint. For example, `https://www.foo.com`.	String	Yes
`fs.oci.client.auth.tenantId`	OCID of your tenancy. To get the value, see Required Keys and OCIDs.	String	Yes
`fs.oci.client.auth.userId`	OCID of the user calling the API. To get the value, see Required Keys and OCIDs.	String	Yes
`fs.oci.client.auth.fingerprint`	Fingerprint for the key pair being used. To get the value, see Required Keys and OCIDs.	String	Yes, unless you provide a custom authenticator.
`fs.oci.client.auth.pemfilepath`	Full path and file name of the private key used for authentication. The file should be on the local file system.	String	Yes, unless you provide a custom authenticator
`fs.oci.client.auth.passphrase`	Passphrase used for the key, if it is encrypted.	String	Only if your key is encrypted

You can specify that a property value applies to a specific bucket by appending `.<bucket_name>.<namespace_name>` to the property name.

This example shows how properties can be configured in a `core-site.xml` file (the OCIDs are shortened for brevity):

```
<configuration>
...
  <property>
    <name>fs.oci.client.hostname</name>
    <value>https://objectstorage.us-ashburn-1.oraclecloud.com</value>
  </property>
  <property>
    <name>fs.oci.client.hostname.myBucket.myNamespace</name>
    <value>https://objectstorage.phoenix-1.oraclecloud.com</value><!-- Use Phoenix for
myBucket@myNamespace -->
  </property>
  <property>
    <name>fs.oci.client.auth.tenantId</name>
    <value>ocid1.tenancy.oc1..aaaaaaaaba3pv6wkcr4j...stifsfdsq</value>
  </property>
  <property>
    <name>fs.oci.client.auth.userId</name>
    <value>ocid1.user.oc1..aaaaaaaat5nvwcnazjc...aqw3rynjq</value>
  </property>
  <property>
    <name>fs.oci.client.auth.fingerprint</name>
    <value>20:3b:97:13:55:1c:5b:0d:d3:37:d8:50:4e:c5:3a:34</value>
  </property>
  <property>
    <name>fs.oci.client.auth.pemfilepath</name>
    <value>~/.oci/oci_api_key.pem</value>
  </property>
...
</configuration>
```

Using Instance Principals for Authentication

Oracle provides *instance principals* so that you no longer need to configure user credentials or provide PEM files on services running on instances. Each of these instances has its own identity and authenticates by using certificates added to the instance by instance principals.

To use instance principals authentication with the HDFS connector, simply provide the property `fs.oci.client.custom.authenticator` and set the value to `com.oracle.bmc.hdfs.auth.InstancePrincipalsCustomAuthenticator`.

Because using instance principals provides your instance with a custom authenticator, it is no long necessary to configure the following properties:

- `fs.oci.client.auth.tenantId`
- `fs.oci.client.auth.userId`
- `fs.oci.client.auth.fingerprint`
- `fs.oci.client.auth.pemfilepath`
- `fs.oci.client.auth.passphrase`

The following example code illustrates using instance principals for authentication with the HDFS connector:

```
<?xml version="1.0"?>
<configuration>
  <property>
    <name>fs.oci.client.hostname</name>
    <value>https://objectstorage.us-phoenix-1.oraclecloud.com</value>
  </property>
  <property>
    <name>fs.oci.client.custom.authenticator</name>
    <value>com.oracle.bmc.hdfs.auth.InstancePrincipalsCustomAuthenticator</value>
  </property>
</configuration>
```

For more information about instance principals, see Announcing Instance Principals for Identity and Access Management.

Configuring a HTTP Proxy

You can set the following optional properties in the `core-site.xml` file to configure a HTTP proxy:

Property	Description	Type	Required
`fs.oci.client.proxy.uri`	The URI of the proxy endpoint. For example, `http://proxy.mydomain.com:80`.	String	No
`fs.oci.client.proxy.username`	The username to authenticate with the proxy.	String	No
`fs.oci.client.proxy.password`	The password to authenticate with the proxy.	String	No
`fs.oci.client.multipart.allowed`	Enables the upload manager to support multipart uploads	Boolean	No
`fs.oci.client.multipart.minobjectsize.mb`	Specifies the minimum object size in mebibytes in order to use the upload manager.	Integer	No
`fs.oci.client.multipart.partsize.mb`	Specifies the part size in mebibytes for the upload manager.	Integer	No



Note

Configuring a proxy enables use of the `ApacheConnectorProvider` when making connections to Object Storage. It buffers requests into memory and can impact memory utilization when uploading large objects. It is recommended to enable multipart uploads and adjust the multipart properties to manage memory consumption.

Large Object Uploads

Large objects are uploaded to Object Storage using multipart uploads. The file is split into smaller parts that are uploaded in parallel, which reduces upload times. This also enables the HDFS connector to retry uploads of failed parts instead of failing the entire upload. However, uploads may transiently fail, and the connector will attempt to abort partially uploaded files. Because these files accumulate (and you will be charged for storage), list the uploads periodically and then after a certain number of days abort them manually using the SDK for Java.

Information about using the Object Storage API for managing multipart uploads can be found in API Documentation.



Note

If you prefer not to use multipart uploads, you can disable them by setting the `fs.oci.client.multipart.allowed` property to `false`.

Best Practices

The following sections contain best practices to optimize usage and performance.

Directory Names

There are no actual directories in Object Storage. Directory grouping is a function of naming convention, where objects use `/` delimiters in their names. For example, an object named `a/example.json` implies there is a directory named `a`. However, if that object is deleted, the `a` directory is also deleted implicitly. To preserve filesystem semantics where the directory can exist without the presence of any files, the HDFS connector creates an actual object whose name ends in `/` with a path that represents the directory, (e.g., create an object named `a/`). Now, deleting `a/example.json` doesn't affect the existence of the `a` directory, because the `a/` object maintains its presence. However, it's entirely possible that somebody could delete that `a/` object without deleting the files/directories beneath it. The HDFS connector will only delete the folder object if there are no objects beneath that path. The folder object itself is zero bytes.

Inconsistent Filesystem

Deleting a directory means deleting all objects that start with the prefix representing that directory. HDFS allows you to query for the file status of a file or a directory. The file status of a directory is implemented by verifying that the folder object for that directory exists. However, it's possible that the folder object has been deleted, but some of the objects with that prefix still exist. For example, in a situation with these objects:

- `a/b/example.json`
- `a/b/file.json`
- `a/b/`

HDFS would know that directory `/a/b/` exists and is a directory, and scanning it would result in `example.json` and `file.json`. However, if object `a/b/` was deleted, the filesystem would appear to be in an inconsistent state. You could query it for all files in directory `/a/b/` and find the two entries, but querying for the status of the actual `/a/b/` directory would result in an exception because the directory doesn't exist. The HDFS connector does not attempt to fix up the state of the filesystem.

FILE CREATION

Object Storage supports objects that can be many gigabytes in size. Creating files will normally be done by writing to a temp file and then uploading the contents of the file when the stream is closed. The temp space must be large enough to handle multiple uploads. The temp directory used is controlled by the `hadoop.tmp.dir` configuration property.

READ/SEEK SUPPORT

When in-memory buffers are enabled (`fs.oci.io.read.inmemory`), seek is fully supported because the entire file is buffered into a byte array. When in-memory buffer is not enabled (likely because object sizes are large), seek is implemented by closing the stream and making a new range request starting at the specified offset.

DIRECTORY LISTING

Listing a directory is essentially a List bucket operation with a prefix and delimiter specified. To create an HDFS FileStatus instance for each key, the connector performs an additional HEAD request to get ObjectMetadata for each individual key. This will be required until Object Storage supports richer list operation data.

URI Format for Filesystems and Files

HDFS filesystems and files are referenced through URIs. The scheme specifies the type of filesystem, and the remaining part of the URI is largely free for the filesystem implementation to interpret as it wants.

Because Object Storage is an object store, its ability to name objects as if they were files in a filesystem is used to mimic an actual filesystem.

ROOT

The root of Object Storage filesystem is denoted by a path where the authority component includes the bucket name and the namespace name, as shown:



Note

In the examples, "`MyBucket`" and "`MyNamespace`" are placeholders and should be replaced with appropriate values.

```
oci://MyBucket@MyNamespace/
```

This is always the root of the filesystem. The reason for using authority for both bucket and namespace is that HDFS only allows the authority portion to determine where the filesystem is; the path portion denotes just the path to the resource (so "oci//MyNamespace/MyBucket" won't work, for example). Note that the `@` character is not a valid character for buckets or namespaces, and should allow the authority to be parsed correctly.

SUB-DIRECTORIES

Sub-directories do not actually exist, but can be mimicked by creating objects with `/` characters. For example, two files named `a/b/c/example.json` and `a/b/d/path.json` would appear as if they were in a common directory `a/b`. This would be achieved by using the Object Storage prefix- and delimiter-based querying. In the given example, referencing a sub-directory as a URI would be:

```
oci://MyBucket@MyNamespace/a/b/
```

OBJECTS/FILES

An object named `a/b/c/example.json` is referenced as:

```
oci://MyBucket@MyNamespace/a/b/c/example.json
```

Logging

Logging in the connector is done through SLF4J. SLF4J is a logging abstraction that allows the use of a user-supplied logging library (e.g., log4j). For more information, see, the SLF4J manual.

The following example shows how to enable basic logging to standard output.

1. Download the SLF4J Simple binding jar: SLF4J Simple Binding
2. Add the jar to your classpath
3. Add the following VM arg to enable debug level logging (by default, info level is used): `-Dorg.slf4j.simpleLogger.defaultLogLevel=debug`

You can configure more advanced logging options by using the log4j binding.

Sample Hadoop Job

hadoop_sample_hdfs:

```
package com.oracle.oci.hadoop.example;

import java.io.ByteArrayOutputStream;
import java.io.IOException;
import java.net.URI;
import java.net.URISyntaxException;

import org.apache.commons.io.IOUtils;
import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.FSDataInputStream;
import org.apache.hadoop.fs.FSDataOutputStream;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;

import com.oracle.oci.hdfs.BmcFilesystem;
```

```
import lombok.RequiredArgsConstructor;

@RequiredArgsConstructor
public class SampleOracleBmcHadoopJob
{
    private static final String SAMPLE_JOB_PATH = "/samplehadoopjob";
    private static final String INPUT_FILE = SAMPLE_JOB_PATH + "/input.dat";
    private static final String OUTPUT_DIR = SAMPLE_JOB_PATH + "/output";

    // non-static since this is the runner class it needs to initialize after we set the properties
    private final Logger log = LoggerFactory.getLogger(SampleOracleBmcHadoopJob.class);

    /**
     * Runner for sample hadoop job. This expects 3 args: path to configuration file, Object Store
namespace, Object
     * Store bucket. To run this, you must:
     *{@code



         *
    Create a standard hadoop configuration file

         *
    Create the bucket ahead of time.

         *}



     * This runner will create a test input file in a file '/samplehadoopjob/input.dat', and job results
will be written
     * to '/samplehadoopjob/output'.
     *
     * @param args
     *            1) path to configuration file, 2) namespace, 3) bucket
     * @throws Exception
     */
    public static void main(final String[] args) throws Exception
    {
        if (args.length != 3)
        {
            throw new IllegalArgumentException(
                    "Must have 3 args: 1) path to config file, 2) object storage namespace, 3) object
```

```
storage bucket");
        }

        // redirect all logs to sysout
        System.setProperty("org.slf4j.simpleLogger.logFile", "System.out");
        System.setProperty("org.slf4j.simpleLogger.defaultLogLevel", "debug");

        final SampleOracleBmcHadoopJob job = new SampleOracleBmcHadoopJob(args[0], args[1], args[2]);
        System.exit(job.execute());
    }

    private final String configurationFilePath;
    private final String namespace;
    private final String bucket;

    public int execute() throws IOException, ClassNotFoundException, InterruptedException,
URISyntaxException
    {
        log.info("Creating hadoop configuration");
        final Configuration configuration = this.createConfiguration(this.configurationFilePath);

        final String authority = this.bucket + "@" + this.namespace;
        final String uri = "oci://" + authority;
        log.info("Using uri: {}", uri);

        log.info("Creating job inputs");
        this.setup(uri, configuration);

        log.info("Creating job");
        final Job job = this.createJob(configuration);

        final String in = uri + INPUT_FILE;
        final String out = uri + OUTPUT_DIR;
        log.info("Using input: {}", in);
        log.info("Using output: {}", out);

        FileInputFormat.addInputPath(job, new Path(in));
        FileOutputFormat.setOutputPath(job, new Path(out));

        log.info("Executing job...");
        final int response = job.waitForCompletion(true) ? 0 : 1;
```

```
        log.info("Attempting to read job results");
        this.tryReadResult(uri, configuration);
        return response;
    }

    private Configuration createConfiguration(final String configFilePath)
    {
        final Configuration configuration = new Configuration();
        configuration.addResource(new Path(configFilePath));
        return configuration;
    }

    private void setup(final String uri, final Configuration configuration) throws IOException,
URISyntaxException
    {
        try (final BmcFilesystem fs = new BmcFilesystem())
        {
            fs.initialize(new URI(uri), configuration);
            fs.delete(new Path(SAMPLE_JOB_PATH), true);
            final FSDataOutputStream output = fs.create(new Path(INPUT_FILE));
            output.writeChars("example\npath\ngak\ntest\nexample\ngak\n\ngak");
            output.close();
        }
    }

    private Job createJob(final Configuration configuration) throws IOException
    {
        final Job job = Job.getInstance(configuration, "word count");
        job.setJarByClass(SampleOracleBmcHadoopJob.class);
        job.setMapperClass(SimpleMapper.class);
        job.setCombinerClass(SimpleReducer.class);
        job.setReducerClass(SimpleReducer.class);
        job.setOutputKeyClass(Text.class);
        job.setOutputValueClass(IntWritable.class);
        return job;
    }

    private void tryReadResult(final String uri, final Configuration configuration)
            throws IOException, URISyntaxException
    {
        try (final BmcFilesystem fs = new BmcFilesystem())
        {
```

```
            fs.initialize(new URI(uri), configuration);
            // this should be the output file name, but that could change
            final FSDataInputStream input = fs.open(new Path(OUTPUT_DIR + "/part-r-00000"));

            final ByteArrayOutputStream baos = new ByteArrayOutputStream();
            IOUtils.copy(input, baos);
            log.info("\n=====\n" + baos.toString() + "=====");
            input.close();
        }
    }
}


package com.oracle.oci.hadoop.example;

import java.io.IOException;
import java.util.StringTokenizer;

import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;

public class SimpleMapper extends Mapper
{
    private final static IntWritable one = new IntWritable(1);
    private final Text word = new Text();

    @Override
    public void map(final Object key, final Text value, final Context context) throws IOException,
InterruptedException
    {
        final StringTokenizer itr = new StringTokenizer(value.toString());
        while (itr.hasMoreTokens())
        {
            this.word.set(itr.nextToken());
            context.write(this.word, one);
        }
    }
}


package com.oracle.oci.hadoop.example;
```

```
import java.io.IOException;

import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Reducer;

public class SimpleReducer extends Reducer
{
    private final IntWritable result = new IntWritable();

    @Override
    public void reduce(final Text key, final Iterable values, final Context context)
            throws IOException, InterruptedException
    {
        int sum = 0;
        for (final IntWritable val : values)
        {
            sum += val.get();
        }
        this.result.set(sum);
        context.write(key, this.result);
    }
}
```

Troubleshooting

This section contains troubleshooting information for the HDFS Connector.

Troubleshooting Service Errors

Any operation resulting in a service error will cause an exception of type com.oracle.bmc.model.BmcException to be thrown by the HDFS Connector. For information about common service errors returned by OCI, see API Errors.

Java Encryption Key Size Errors

The HDFS Connector can only handle keys of 128 bit or lower key length. Users get "Invalid Key Exception" and "Illegal key size" errors when they use longer keys, such as AES256. Use one of the following workarounds to fix this issue:

- Use a 128 bit key, such as AES128.
- Install the Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction from the following location: http://www.oracle.com/technetwork/java/javase/downloads/jce8-download-2133166.html

Contributions

Got a fix for a bug, or a new feature you'd like to contribute? The SDK is open source and accepting pull requests on GitHub.

Notifications

If you wish to be notified when a new version of the HDFS connector is released, subscribe to the Atom feed.

Questions or Feedback

Ways to get in touch:

- GitHub Issues: To file bugs and make feature requests
- Stack Overflow: Please use the oracle-cloud-infrastructure and oci-hdfs-connector tags in your post
- Oracle Cloud Infrastructure Forum: Threads tagged with Developer Tools
- My Oracle Support

Using the HDFS Connector with Spark

Introduction

This article provides a walkthrough that illustrates using the HDFS connector with the Spark application framework. For the walkthrough, we use the Oracle Linux 7.4 operating system, and we run Spark as a standalone on a single computer.

PREREQUISITES

Following are prerequisites for completing the walkthrough:

- You must have permission to launch a Compute instance. For guidance, see Launching an Instance.
- You must be able to connect to the service instance that you've launched. For guidance, see Connecting to an Instance.
- You must have the appropriate OCID, fingerprint, and private key for the Identity and Access Management (IAM) user that you will use to interact with an Object Storage. For guidance, see SDK and Tool Configuration; see also Resource Identifiers.
- You must have an Object Storage bucket that you can connect to.
- The IAM user must be able to read and write to that bucket using the Console.

USING SPARK

INSTALL SPARK AND DEPENDENCIES



Note

For the purpose of this example, install Spark into the current user's home directory. Note that for production scenarios, you would not do this.

1. Launch an instance of your Compute service. For guidance, see Launching an Instance.
2. Ensure that your service instance has a public IP address so that you can connect using a Secure Shell (SSH) connection. For guidance, see Instance Console Connections.
3. Connect to your service instance using an SSH connection.
4. Install Spark and its dependencies, Java and Scala, by using the code examples that follow.

```
# We'll use wget to download some of the artifacts that need to be installed
sudo yum install wget
```

```
# First install Java
sudo yum install java-1.8.0-openjdk.x86_64
export JAVA_HOME=/usr/lib/jvm/jre-1.8.0-openjdk
# Should be something like: OpenJDK Runtime Environment (build 1.8.0_161-b14)
java -version

# Then install Scala
wget https://downloads.lightbend.com/scala/2.12.4/scala-2.12.4.rpm
sudo yum install scala-2.12.4.rpm
# Should be something like: Scala code runner version 2.12.4 -- Copyright 2002-2017, LAMP/EPFL and
Lightbend, Inc.
scala -version

# Then download Spark
wget https://archive.apache.org/dist/spark/spark-2.2.1/spark-2.2.1-bin-hadoop2.7.tgz
tar xvf spark-2.2.1-bin-hadoop2.7.tgz
export SPARK_HOME=$HOME/spark-2.2.1-bin-hadoop2.7
export PATH=$PATH:$SPARK_HOME/bin

# Start a Spark master
cd $SPARK_HOME
./sbin/start-master.sh
```

DOWNLOAD THE HDFS CONNECTOR AND CREATE CONFIGURATION FILES



Note

For the purposes of this example, place the JAR and key files in the current user's home directory. For production scenarios you would instead put these files in a common place that enforces the appropriate permissions (that is, readable by the user under which Spark and Hive are running).

Download the HDFS connector to the service instance and add the relevant configuration files by using the following code example. For additional information, see HDFS Connector for Object Storage.

```
wget https://docs.cloud.oracle.com/tools/hdfs/latest/download/oci-hdfs.zip
unzip oci-hdfs.zip -d oci-hdfs

cd $HOME
mkdir .oci
# Create or copy your API key into the $HOME/.oci directory

cd $SPARK_HOME/conf
# Create a core-site.xml (e.g. by transferring one you have, using vi etc.). Consult
# https://docs.cloud.oracle.com/Content/API/SDKDocs/hdfsconnector.htm#two
# for what this should look like

# Create a spark-defaults.conf file from the template
cp spark-defaults.conf.template spark-defaults.conf
```

In the `spark-defaults.conf` file, add the following at the bottom:

```
spark.sql.hive.metastore.sharedPrefixes= shaded.oracle,com.oracle.bmc
```

PREPARE DATA

For testing data, we will use the MovieLens data set.

1. Download the latest data set at https://grouplens.org/datasets/movielens/latest/. Be sure to download the "Small" data set.
2. Unzip the download file.
3. Upload the `movies.csv` file to your Object Storage bucket.

TEST USING THE SPARK SHELL

With the data ready, we can now launch the Spark shell and test it using a sample command:

```
cd $SPARK_HOME
./bin/spark-shell

scala> sc.wholeTextFiles("oci://PipedUploadTest@internalbriangustafson/")
java.io.IOException: No FileSystem for scheme: oci
```

You receive an error at this point because the oci:// file system schema is not available. We need to reference the JAR file before starting the Spark shell. Here's an example for doing so:

```
./bin/spark-shell --jars $HOME/oci-hdfs/lib/oci-hdfs-full-1.2.7.jar

scala> sc.wholeTextFiles("oci://PipedUploadTest@internalbriangustafson/")
res0: org.apache.spark.rdd.RDD[(String, String)] = oci://PipedUploadTest@internalbriangustafson/
MapPartitionsRDD[1] at wholeTextFiles at <console>:25

scala> sc.textFile("oci://PipedUploadTest@internalbriangustafson/movies.csv").take(20).foreach
(println)
movieId,title,genres
1,Toy Story (1995),Adventure|Animation|Children|Comedy|Fantasy
2,Jumanji (1995),Adventure|Children|Fantasy
3,Grumpier Old Men (1995),Comedy|Romance
4,Waiting to Exhale (1995),Comedy|Drama|Romance
5,Father of the Bride Part II (1995),Comedy
6,Heat (1995),Action|Crime|Thriller
7,Sabrina (1995),Comedy|Romance
8,Tom and Huck (1995),Adventure|Children
9,Sudden Death (1995),Action
10,GoldenEye (1995),Action|Adventure|Thriller
11,"American President, The (1995)",Comedy|Drama|Romance
12,Dracula: Dead and Loving It (1995),Comedy|Horror
13,Balto (1995),Adventure|Animation|Children
14,Nixon (1995),Drama
15,Cutthroat Island (1995),Action|Adventure|Romance
16,Casino (1995),Crime|Drama
17,Sense and Sensibility (1995),Drama|Romance
18,Four Rooms (1995),Comedy
19,Ace Ventura: When Nature Calls (1995),Comedy
```

The command is successful so we are able to connect to Object Storage. Note that if you do not wish to pass the `--jars` argument each time the command executes, you can instead copy the `oci-hdfs-full` JAR file into the `$SPARK_HOME/jars` directory.

START THE SPARK THRIFT SERVER

Start the Spark Thrift Server on port 10015 and use the Beeline command line tool to establish a JDBC connection and then run a basic query, as shown here:

```
cd $SPARK_HOME
./sbin/start-thriftserver.sh --hiveconf hive.server2.thrift.port=10015
```

Once the Spark server is running, we can launch Beeline, as shown here:

```
cd $SPARK_HOME
./bin/beeline
Beeline version 1.2.1.spark2 by Apache Hive
beeline>
```

Next, connect to the server, as shown here:



Note

For the purposes of this example, we have not configured any security, so any user name and password will be accepted. For production scenarios you would not do this.

```
beeline> !connect jdbc:hive2://localhost:10015 testuser testpass
Connecting to jdbc:hive2://localhost:10015
log4j:WARN No appenders could be found for logger (org.apache.hive.jdbc.Utils).
log4j:WARN Please initialize the log4j system properly.
log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more info.
Connected to: Spark SQL (version 2.2.1)
Driver: Hive JDBC (version 1.2.1.spark2)
Transaction isolation: TRANSACTION_REPEATABLE_READ
0: jdbc:hive2://localhost:10015>
```

If we now check to see what tables exist, we see the following:

```
0: jdbc:hive2://localhost:10015> show tables;
+-----------+------------+--------------+--+
| database  | tableName  | isTemporary  |
+-----------+------------+--------------+--+
+-----------+------------+--------------+--+
No rows selected (0.724 seconds)
```

None exist presently; however, we can create a talbe and link it to the `movies.csv` file that we downloaded and placed in the Object Storage bucket, as shown here:

```
0: jdbc:hive2://localhost:10015> create table test_table (movieId integer, title string, genres
string) using csv options (path "oci://myBucket@myTenant/movies.csv", header "true", delimiter ",");

0: jdbc:hive2://localhost:10015> describe formatted test_table;
+-------------------------------+----------------------------------------------------------------+------
```

```
----+--+
|           col_name            |                        data_type                         |
comment  |
+-------------------------------+----------------------------------------------------------+------
----+--+
| movieId                       | int                                                      | NULL
   |
| title                         | string                                                   | NULL
   |
| genres                        | string                                                   | NULL
   |
|                               |                                                          |
   |
| # Detailed Table Information  |                                                          |
   |
| Database                      | default                                                  |
   |
| Table                         | test_table                                               |
   |
| Owner                         | opc                                                      |
   |
| Created                       | Thu Mar 01 20:45:18 GMT 2018                             |
   |
| Last Access                   | Thu Jan 01 00:00:00 GMT 1970                             |
   |
| Type                          | EXTERNAL                                                 |
   |
| Provider                      | csv                                                      |
   |
| Table Properties              | [transient_lastDdlTime=1519937118]                       |
   |
| Location                      | oci://PipedUploadTest@internalbriangustafson/movies.csv  |
   |
| Serde Library                 | org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe       |
   |
| InputFormat                   | org.apache.hadoop.mapred.SequenceFileInputFormat         |
   |
| OutputFormat                  | org.apache.hadoop.hive.ql.io.HiveSequenceFileOutputFormat  |
   |
| Storage Properties            | [delimiter=,, header=true, serialization.format=1]       |
   |
+-------------------------------+----------------------------------------------------------+------
```

```
----+--+
```

Note that the table stores its data externally in Object Storage and the data can be accessed using the HDFS connector (the `oci://` file system scheme). Now that we have a table, we can query it:

```
0: jdbc:hive2://localhost:10015> select * from test_table limit 10;
+----------+--------------------------------------+-------------------------------------------+--+
| movieId  |                title                 |                  genres                   |
+----------+--------------------------------------+-------------------------------------------+--+
| 1        | Toy Story (1995)                     | Adventure|Animation|Children|Comedy|Fantasy  |
| 2        | Jumanji (1995)                       | Adventure|Children|Fantasy                    |
| 3        | Grumpier Old Men (1995)              | Comedy|Romance                               |
| 4        | Waiting to Exhale (1995)             | Comedy|Drama|Romance                         |
| 5        | Father of the Bride Part II (1995)   | Comedy                                    |
| 6        | Heat (1995)                          | Action|Crime|Thriller                      |
| 7        | Sabrina (1995)                       | Comedy|Romance                               |
| 8        | Tom and Huck (1995)                  | Adventure|Children                          |
| 9        | Sudden Death (1995)                  | Action                                    |
| 10       | GoldenEye (1995)                     | Action|Adventure|Thriller                   |
+----------+--------------------------------------+-------------------------------------------+--+
```

FOR MORE INFORMATION

- HDFS Connector for Object Storage
- Overview of Object Storage
- Apache Spark

Terraform Provider

This topic provides information about installing, configuring, and using the Terraform provider with Oracle Cloud Infrastructure.

Terraform is a tool that allows you to programmatically manage, version, and persist your IT infrastructure as "infrastructure as code." Terraform uses declarative syntax to describe your infrastructure and then persist it in configuration files that can be shared, reviewed, edited, versioned, preserved, and reused.

The Oracle Cloud Infrastructure Terraform provider is a component that connects Terraform to the service infrastructure that you wish to manage.

- **Services supported:**
  - Audit
  - Container Engine for Kubernetes
  - Compute Autoscaling
  - Core Services (Networking, Compute, Block Volume)
  - Database
  - DNS Service
  - Email Delivery
  - FastConnect
  - File Storage
  - Health Checks
  - IAM
  - Key Management
  - Load Balancing
  - Monitoring
  - Notifications
  - Object Storage
  - Streaming
  - Web Application Firewall (WAF)
- **Licensing:** This provider and sample is licensed under the Mozilla Public License 2.0; third-party content is separately licensed as described in the code.
- **Documentation:** Oracle Cloud Infrastructure Provider

Contributions

Got a fix for a bug, or a new feature you'd like to contribute? The Terraform provider is open source and accepting pull requests on GitHub.

Notifications

To be notified when a new version of the Terraform provider is released, subscribe to the Atom feed.

Questions or Feedback

Ways you can get in touch:

- GitHub: To file bugs and feature requests only.
- Stack Overflow: Please use the oci-terraform and oracle-cloud-infrastructure tags in your post.
- Developer Tools section of the Oracle Cloud forums.

Getting Started with the Terraform Provider

This topic provides instructions for downloading and installing both Terraform and the Oracle Cloud Infrastructure Terraform provider, and provides a brief introduction to the key concepts for understanding and using the Oracle Cloud Infrastructure Terraform provider.

Terraform Overview

Terraform is "infrastructure-as-code" software that allows you to define your infrastructure resources in files that you can persist, version, and share. These files describe the steps required to provision your infrastructure and maintain its desired state; it then executes these steps and builds out the described infrastructure.

Terraform’s configuration and execution building blocks are modules, which are self-contained configuration packages. You can use these modules to organize your code and to create reusable components. HashiCorp, the developer of Terraform, provides a library of open-source Terraform modules “out of the box” to support many common tasks.

To use Terraform for Oracle Cloud Infrastructure, you must download two components - Terraform from HashiCorp, and then the Oracle Cloud Infrastructure Terraform provider.

Download and Install Terraform

Download Terraform from the HashiCorp download page. Ensure that you download the correct binary file for your system.



Important

The Oracle Cloud Infrastructure Terraform provider version 2.2.0 and greater requires Terraform version 0.10.1 or greater.

Download the Oracle Cloud Infrastructure Terraform Provider

Prerequisites for Installing and Using the Terraform Provider

- An Oracle Cloud Infrastructure account that has user credentials sufficient to execute a Terraform plan.
- A user in that account.
- Required keys and Oracle Cloud Infrastructure IDs (OCIDs). For guidance, see "Required Keys and OCIDs" in the *Oracle Cloud Infrastructure User Guide*.
- The correct Terraform binary file for your operating system (version 0.10.1 or greater).

Installing and Configuring the Terraform Provider

- For guidance on installing or on upgrading a previous version of the Oracle Cloud Infrastructure Terraform provider, see Terraform Provider Version 3.
- For guidance on setting up the Terraform provider, see Oracle Cloud Infrastructure Provider.

FOR MORE INFORMATION

- GitHub
- Hashicorp Terraform Documentation
- Creating Terraform Modules
- Terraform Configurations
- Terraform Configuration Syntax

Writing Terraform Configurations

OVERVIEW

Using Terraform, you can describe your Oracle Cloud Infrastructure using the HashiCorp Configuration Language format (HCL) in Terraform configuration files (see Configuration Syntax). Terraform configuration files can use either of two formats: Terraform domain-specific language (HashiCorp Configuration Language format [HCL]), which is the recommended approach, or JSON format if the files need to be machine-readable. Configuration files that use the HCL format end with the `.tf` file extension; those using JSON format end with the `.tf.json` file extension. The Terraform format is human-readable, while the JSON format is machine readable.

Use Terraform configurations to define your Oracle Cloud Infrastructure resources, variable definitions, data sources, and a great deal more. Terraform, then, converts your Oracle Cloud Infrastructure configurations into a set of API calls against Oracle Cloud Infrastructure API endpoints. The key to writing Terraform configuration is understanding how to abstract the desired infrastructure conceptually into Terraform configuration syntax.



Important

While the Oracle Cloud Infrastructure API uses camelCase extensively, Terraform does not support camelCase in configuration files. For this reason, in the configurations you see underscores rather than capitalization as separators. For example, where the API uses `availabilityDomain`, the Terraform configuration uses `availability_domain`.

CONFIGURATION FILE REQUIREMENTS

Terraform configuration (`.tf`) files have specific requirements, depending on the components that are defined in the file. For example, you might have your Terraform provider defined in one file (provider.tf), your variables defined in another (variables.tf), your data sources defined in yet another.



Note

We provide a great many example configuration files in the Terraform Provider Examples on our Oracle Cloud Infrastructure GitHub.

PROVIDER DEFINITIONS

The following example using Terraform syntax illustrates the requirements for an Oracle Cloud Infrastructure Terraform provider definition, and also shows associated variable definitions. The provider definition relies on variables so that the configuration file itself does not contain sensitive data. Including sensitive data creates a security risk when exchanging or sharing configuration files.

```
variable "tenancy_ocid" {}
variable "user_ocid" {}
variable "fingerprint" {}
```

```
variable "private_key_path" {}
variable "region" {}

provider "oci" {
   tenancy_ocid = "${var.tenancy_ocid}"
   user_ocid = "${var.user_ocid}"
   fingerprint = "${var.fingerprint}"
   private_key_path = "${var.private_key_path}"
   region = "${var.region}"
}
```

The `region` attribute specifies the geographical region in which your provider resources are created. To target multiple regions in a single configuration, you simply create a provider definition for each region and then differentiate by using a provider alias, as shown in the following example. Notice that only one provider, named "oci" is defined, and yet the `oci` provider definition is entered twice, once for the `us-phoenix-1` region (with the alias "phx"), and once for the region `us-ashburn-1` (with the alias "iad").

```
variable "tenancy_ocid" {}
variable "user_ocid" {}
variable "fingerprint" {}
variable "private_key_path" {}
variable "compartment_ocid" {}

provider "oci" {
   region = "us-phoenix-1"
   alias = "phx"
   tenancy_ocid = "${var.tenancy_ocid}"
   user_ocid = "${var.user_ocid}"
   fingerprint = "${var.fingerprint}"
   private_key_path = "${var.private_key_path}"
}

provider "oci" {
   region = "us-ashburn-1"
   alias = "iad"
   tenancy_ocid = "${var.tenancy_ocid}"
   user_ocid = "${var.user_ocid}"
   fingerprint = "${var.fingerprint}"
   private_key_path = "${var.private_key_path}"
}
```

For more information, see Provider Configuration.

Variable Definitions

Variables in Terraform represent parameters for Terraform modules. In variable definitions, each block configures a single input variable, and each definition can take any or all of three optional arguments:

- `type` (optional): Defines the variable type as one of three allowed values: `string`, `list`, and `map`. If this argument is not used, the variable type is inferred based on `default`. If no `default` is provided, the type is assumed to be `string`.
- `default` (optional): Sets the default value for the variable. If no default value is provided, the caller must provide a value or Terraform throws an error.
- `description` (optional): A human-readable description of the variable.

Following are examples of several variable definitions. Some definitions include optional parameters.

```
variable "tenancy_ocid" {}
variable "user_ocid" {}
variable "fingerprint" {}
variable "private_key_path" {}
variable "region" {}

variable "AD" {
    default     = "1"
    description = "Availability Domain"
}

variable "CPUCoreCount" {
    default = "2"
    type    = "string"
}
```

For more information, see Input Variable Configuration. See also Input Variables.

Output Configuration

Output variables provide a means to support Terraform end-user queries. This allows users to extract meaningful data from among the potentially massive amount of data associated with a complex infrastructure. For example, you might be interested only in a handful of key values

at any given time and defining output variables allows you to extract exactly the information that you need.

Following is a simple example in which only a few output variables (instance IP addresses and boot volume IDs) are defined:

```
# Output the private and public IPs of the instance
output "InstancePrivateIPs" {
value = ["${oci_core_instance.TFInstance.*.private_ip}"]
}

output "InstancePublicIPs" {
value = ["${oci_core_instance.TFInstance.*.public_ip}"]
}

# Output the boot volume IDs of the instance
output "BootVolumeIDs" {
  value = ["${oci_core_instance.TFInstance.*.boot_volume_id}"]
}
```

For more information, see Output Variables. See also Output Configuration.

RESOURCE CONFIGURATION

Resources are components of your Oracle Cloud Infrastructure. These resources include everything from low-level components such as physical and virtual servers, to higher-level components such as email and database providers, your DNS record.

Following is a simple example of a resource definition that illustrates their basic structure.

```
resource "oci_core_virtual_network" "vcn1" {
   cidr_block = "10.0.0.0/16"
   dns_label = "vcn1"
   compartment_id = "${var.compartment_ocid}"
   display_name = "vcn1"
}
```

The resource declaration on the first line of the example uses the keyword "resource" and takes two parameters, resource `type` and resource `name` ("oci_core_virtual_network" and "vcn1" in the example). Inside the code block, then, is the resource configuration.

For more information, see Resource Configuration.

DATA SOURCE CONFIGURATION

Data sources represent read-only views of existing infrastructure intended for semantic use in Terraform configurations. Following is a simple example of a data source configuration to illustrate its basic structure:

```
# Gets a list of Availability Domains
data "oci_identity_availability_domains" "ADs" {
  compartment_id = "${var.tenancy_ocid}"
}

# Get DB node list
data "oci_database_db_nodes" "DBNodeList" {
  compartment_id = "${var.compartment_ocid}"
  db_system_id = "${oci_database_db_system.TFDBNode.id}"
}

# Get DB node details
data "oci_database_db_node" "DBNodeDetails" {
  db_node_id = "${lookup(data.oci_database_db_nodes.DBNodeList.db_nodes[0], "id")}"
}

# Gets the OCID of the first (default) vNIC
data "oci_core_vnic" "DBNodeVnic" {
  vnic_id = "${data.oci_database_db_node.DBNodeDetails.vnic_id}"
}
```

For more information, see Data Source Configuration.

ENABLING INSTANCE PRINCIPAL AUTHORIZATION

Instance principal authorization allows your provider to make API calls from an Oracle Cloud Infrastructure compute instance without needing the `tenancy_ocid`, `user_ocid`, `fingerprint`, and `private_key_path` attributes in your provider definition.



Note

Instance principle authorization applies only to instances that are running in the Oracle Cloud Infrastructure.

To enable instance principal authorization for Oracle Cloud Infrastructure Terraform providers, set the `auth` attribute to "InstancePrincipal" in your provider definition, as shown in the following example:

```
variable "region" {}

provider "oci" {
   auth = "InstancePrincipal"
   region = "${var.region}"
}
```

Example Terraform Providers

To see examples of the Oracle Cloud Infrastructure Terraform provider, see Terraform Provider Examples. Several examples are provided and are grouped by service, including Compute, Database, Networking, Load Balancing, and several others.

For More Information

- Configuration
- Configuration Syntax
- Terraform Provider Examples

Using the Object Store for Terraform State Files

You can store Terraform state files in the Oracle Cloud Infrastructure Object Storage. Doing so requires that you configure a backend using one of the Terraform backend types.

Terraform supports various backend types to allow flexibility in how state files are loaded into Terraform. (For more information, see Terraform Backend Types.) For our purposes, we address two of these approaches:

- Using an HTTP remote state backend
- Using an S3-compatible remote state backend

USING AN HTTP BACKEND

Using the HTTP backend type allows you to store state using a simple REST client. With the HTTP backend type, you can easily fetch, update, and purge state using the `HTTP GET`, `POST`, and `DELETE` methods.

To configure the HTTP backend to store your Oracle Cloud Infrastructure Terraform state files, do the following:

CREATE A PRE-AUTHENTICATED REQUEST

Creating a pre-authenticated request in Oracle Object Storage enables accessing a bucket or object in the Oracle Cloud Infrastructure without needing to provide credentials. To do so, you must create a pre-authenticated request that has read/write permissions to the object store where you intend to save the Terraform state file. You can do so in any of three ways: by using the Console UI, by using the command line interface (CLI), or by using the REST APIs.



Note

A state file must exist in the bucket before you create the pre-authenticated request. This file can be an existing state file, or an empty file for the initial state.

For guidance, see Using Pre-Authenticated Requests.

UPLOAD EXISTING STATE

If you have an existing state file, you can upload it using Curl to make an `HTTP Put` request to the object store URL, as shown here:

```
curl -X PUT -H "Content-Type: text/plain" --data-binary "@path/to/local/tfstate" http://<prefix>/<my-access-uri>
```

CONFIGURE HTTP AS A TERRAFORM BACKEND

The HTTP backend type stores state using a simple REST client and allows you to easily fetch, update, and purge state using the `HTTP GET`, `POST`, and `DELETE` methods.

The access URI for addressing Oracle Cloud Infrastructure Terraform configurations must be of the form : `https://objectstorage.us-phoenix-1.oraclecloud.com/my-access-uri` (where region and access URI are specific to you).

For more example configuration and state files that reference code, and a summary of configuration variables, see Standard Backends: HTTP.

Following is an example Terraform configuration. The region in the URL can be something other than the Phoenix region.

```
terraform {
   backend "http" {
     address = "https://objectstorage.us-phoenix-1.oraclecloud.com/<my-access-uri>" update_method = "PUT" }
}
```

REINITIALIZE TERRAFORM

Finally, you must reinitialize Terraform and then run the `apply` command, as shown following.

```
terraform init
terraform apply
```

After completing these steps,, you are able to use Oracle Cloud Infrastructure as the backend for storing Terraform state files.

USING AN S3-COMPATIBLE BACKEND

Configuring the S3-compatible backend requires that the account be enabled with S3 authentication keys, which are set on a per-user basis.

1. In the Console, open the navigation menu, then, under **Governance and Administration**, navigate to **Identity**, then **Users**. Under **User Details**, click **Customer Secret Key**. For guidance, see Working with Amazon S3 Compatibility API Keys.
2. Set the location for the credentials file. The default location is `~/.aws/credentials`. You can set an alternate location by using the S3 backend `shared_credentials_file` option.



Warning

Never set the `access_key` and the `secret_key` attributes in the same Terraform backend configuration, since this creates a security risk.

3. Configure the `[default]` entry in the credentials file with the appropriate object storage credentials. The file can contain any number of credential profiles. If you provide a different profile name, you must also update the backend `profile` option in your Terraform configuration file.

   Following is an example of Object Storage credentials:

   ```
   [default]
   aws_access_key_
   id=ocid1.credential.oc1..aaaaaaaasbmhehdmefolvqwtbdjgwfsxjsgxgipdbph7odn2brgurgsyztca
   aws_secret_access_key=mSTdaWhlbWj3ty4JZXlm0NUZV52xlImWjayJLJ6OH9A=
   ```

   Where `aws_access_key_id` and `aws_secret_access_key` are user-specific values provided from the Console. The key values provided in the example are not valid and provided as examples only.

4. Set the object storage `endpoint` value in the following format:

   `https://{tenancy}.compat.objectstorage.{region}.oraclecloud.com`

Following is a full example of an Object Storage backend configuration:

```
terraform {
  backend "s3" {
    bucket    = "terraform-state"
```

```
    key      = "terraform.tfstate"
    region   = "us-phoenix-1"
    endpoint = "https://acme.compat.objectstorage.us-phoenix-1.oraclecloud.com"

    skip_region_validation      = true
    skip_credentials_validation = true
    skip_requesting_account_id  = true
    skip_get_ec2_platforms      = true
    skip_metadata_api_check     = true
    force_path_style            = true
  }
}
```



Note

The S3 backend configuration can also be used for the `terraform_remote_state` data source to enable sharing state across Terraform projects.

Once you have configured the backend, you must run `terraform init` to finish the setup. If you already have an existing `terraform.tfstate` file, then Terraform prompts you to confirm that the current state file is the one to upload to the remote state.

For More Information

- Using Pre-Authenticated Requests
- State Files
- Terraform Backend Types

Terraform Provider Best Practices

Following are recommended best practices for writing configurations for the Oracle Cloud Infrastructure Terraform provider.

- Referencing Images
- Availability Domains

REFERENCING IMAGES

When launching Compute instances, your Terraform configuration should use the same image every time you run a Terraform Apply job.

To ensure this, specify the image OCID directly, rather than locating it using the `oci_core_image` data source. This is because the `oci_core_image` data source calls into the `ListImages` API, whose return values can change over time because over time images are added and older ones deleted. For a list of Oracle-Provided images and their OCIDs, see Oracle-Provided Images. For more information, see Results of oci_core_images will change over time for Oracle-provided images.

We recommend the following pattern for specifying an image for a given region:

```
variable "image_id" {
  type = "map"
  default = {
    // See https://docs.cloud.oracle.com/iaas/images/
    // Oracle-provided image "Oracle-Linux-7.4-2018.02.21-1"
    us-phoenix-1 = "ocid1.image.oc1.phx.aaaaaaaaupbfz5f5hdvejulmalhyb6goieolullgkpumorbvxlwkaowglslq"
    us-ashburn-1 = "ocid1.image.oc1.iad.aaaaaaaajlw3xfie2t5t52uegyhiq2npx7bqyu4uvi2zyu3w3mqayc2bxmaa"
    eu-frankfurt-1 = "ocid1.image.oc1.eu-frankfurt-
1.aaaaaaaa7d3fsb6272srnftyi4dphdgfjf6gurxqhmv6ileds7ba3m2gltxq"
    uk-london-1 = "ocid1.image.oc1.uk-london-
1.aaaaaaaaa6h6gj6v4n56mqrbgnosskq63blyv2752g36zerymy63cfkojiiq"
  }
}
```

A Compute instance can use this in the following way:

```
resource "oci_core_instance" "TFInstance" {
  image = "${var.image_id[var.region]}"
  ...
}
```

AVAILABILITY DOMAINS

With respect to Availability Domains, we caution against a common pattern, as shown here:

```
// Get all availability domains for the region
data "oci_identity_availability_domains" "ads" {
  compartment_id = "${var.tenancy_ocid}"
}
```

```
// Then either use it to get a single AD name based on the index:
resource "oci_core_instance" "nat" {
  availability_domain = "${lookup(data.oci_identity_availability_domains.ads.availability_domains
[var.nat_instance_ad],"name")}"
  ...
}

// Or iterate through all the ADs:
resource "oci_core_subnet" "nat" {
  count = "${length(data.oci_identity_availability_domains.ads.availability_domains)}"
  availability_domain = "${lookup(data.oci_identity_availability_domains.ad.availability_domains
[count.index], "name")}"
  ...
}
```

The recommendation, then, is to explicitly list the Availability Domain names for the regions in your configuration. To do so, use a variable that you have defined as follows:

```
variable "ad_list" {
  type = "list"
}
```

You can then use the variable as shown here:

```
// Index:
resource "oci_core_instance" "nat" {
  availability_domain = "${var.ad_list[var.nat_instance_ad_index]}"
  ...
}

// Or iterate through all the ADs:
resource "oci_core_subnet" "nat" {
  count = "${length(var.ad_list)}"
  availability_domain = "${var.ad_list[count.index]}"
  ...
}
```

You can then set the `ad_list` variable directly by using the availability domain names for your tenant and region, as shown here:

```
variable "ad_list" {
  type = "list"
```

```
  default = ["kIdk:PHX-AD-1","kIdk:PHX-AD-2","kIdk:PHX-AD-3"]
}
```

The advantage of using this method is that it gives you control over your availability domain usage and prevents unexpected changes over time. However, this approach is problematic when configurations are shared between tenancies and regions, since availability domain names are tenancy- and region-specific.

A convenient alternative is to instead set the `ad_list` value by using the `oci_identity_availability_domains` data source. You should do this in the configuration, then pass them into the modules. This effectively centralizes the list of ADs, making it is easy to switch to an explicit list later, should that become necessary: Note that the modules themselves should not use the `oci_identity_availability_domains` data source.

```
data "oci_identity_availability_domains" "ad" {
  compartment_id = "${var.tenancy_ocid}"
}

data "template_file" "ad_names" {
  count = "${length(data.oci_identity_availability_domains.ad.availability_domains)}"
  template = "${lookup(data.oci_identity_availability_domains.ad.availability_domains[count.index],
"name")}"
}

module "ssm_network" {
  ad_list = "${data.template_file.ad_names.*.rendered}"
  ...
}
```

Ansible Modules

This topic provides information about installing, configuring, and using Ansible and the Oracle Cloud Infrastructure Ansible modules.

Oracle supports the use of Ansible for cloud infrastructure provisioning, orchestration, and configuration management. Ansible allows you to automate configuring and provisioning your

cloud infrastructure, deploying and updating software assets, and orchestrating your complex operational processes.

What enables orchestrating, provisioning, and configuration management tasks are the Ansible modules for Oracle Cloud Infrastructure. Ansible provides a library of these Ansible modules "out of the box" for managing common tasks, and libraries of custom modules from cloud providers like AWS and Azure (see the Module Index). Oracle also provides a library of Ansible cloud modules that support provisioning and managing Oracle Cloud Infrastructure services.

Ansible playbooks automate configuration, deployment, and orchestration tasks using a declarative language (YAML), which allows you to describe infrastructure configuration, deployment policy, and orchestrating complex process steps, either synchronously or asynchronously. Ansible playbooks can be thought of as automation instruction manuals; Ansible modules, then, are your task execution tools.

Ansible modules allow you to author Ansible playbooks that enable automating the provisioning and configuring of Oracle Cloud Infrastructure services and resources, such as Compute, Load Balancing, Database, and other Oracle Cloud Infrastructure services.

- **Services supported:**
  - Block Volume
  - Compute
  - Container Engine for Kubernetes
  - Database
  - DNS
  - Email Delivery
  - File Storage
  - IAM
  - Load Balancing
  - Networking
  - Object Storage

    - Search
    - WAF
- **Licensing:** Copyright © 2018, Oracle and/or its affiliates. This software is made available to you under the terms of the GPL 3.0 license or the Apache 2.0 license. See LICENSE.TXT for details.
- **Download:** You can download the Oracle Cloud Infrastructure Ansible Module from the Oracle GitHub repository. Please follow guidance in Getting Started with Ansible for Oracle Cloud Infrastructure, in the section "Installing the Oracle Cloud Infrastructure Ansible Modules."
- **Documentation:** Getting Started with Ansible for Oracle Cloud Infrastructure

Requirements

To use Ansible, you must have the following prerequisites on your controller computer, that is, the computer from which Ansible playbooks are executed. For more information, see the Ansible Installation Guide.

- An Oracle Cloud Infrastructure account.
- A user created in that account, in a security group with a policy that grants the necessary permissions for working with resources in those compartments. For guidance, see How Policies Work.
- The necessary credentials and OCID information.
- The Oracle Cloud Infrastructure Python SDK. To download and install the SDK, see the topic Python SDK.
- Install the Ansible modules by following guidance in Getting Started with Ansible for Oracle Cloud Infrastructure, in the section "Installing the Oracle Cloud Infrastructure Ansible Modules."

Ansible Key Concepts

MODULES

Modules represent discrete provisioning tasks or operations that you can invoke individually from the command line, or else run individually or in sequence from a playbook. Ansible provides a large library of out-of-box modules that are listed in the Module Index. Other Ansible providers like Microsoft Azure and Amazon Web Services (AWS) also provide libraries of Ansible modules. Oracle also provides a library of Ansible modules that interact with services. For more information, see Working with Modules.

PLAYBOOKS

Playbooks provide a declarative language that allows you to create and persist your Ansible cloud infrastructure provisioning tasks. Playbooks are coded sets of instructions that you create to manage cloud infrastructure provisioning, and more advanced processes. For more information, see Working with Playbooks. See also a set of available Example Playbooks.

ROLES

Ansible roles are units of organization that allows you to abstract configuration, orchestration, and provisioning tasks into roles that you can save and share among playbooks and other users, and that are useful for organizing functionality in playbooks. In a sense, playbooks are minimalist playbooks that encapsulate common configuration steps so you can share them across users or playbooks. For more information, see Ansible Roles.

INVENTORY

Ansible inventory is a means for describing hosts and groups. The inventory can be static (as a simple .ini file), or dynamic, where a look-up script assembles an up-to-date infrastructure inventory. For more information about Ansible inventory, see Working with Inventory.



Important

When using Ansible to work with Oracle Cloud Infrastructure hosts, we recommend using the Oracle dynamic inventory script to obtain a dynamic inventory list. For more information, see Using the Dynamic Inventory Script.

Notifications

To be notified when new Ansible modules are released, subscribe to the Ansible Atom Feed.

Questions or Feedback

Ways to get in touch:

- GitHub: To file bugs and feature requests only.
- Stack Overflow: Use the oci-ansible and oracle-cloud-infrastructure tags in your post.
- Developer Tools section of the Oracle Cloud forums.

Getting Started with Ansible for Oracle Cloud Infrastructure

This topic discusses how to get started with downloading and using Ansible with the Oracle Cloud Infrastructure. There are four initial steps for getting started with Ansible:

- Ensure that you have all of the prerequisites
- Download and install the Oracle Cloud Infrastructure Python SDK
- Download and install Ansible
- Download and install the Ansible modules for Oracle Cloud Infrastructure

Prerequisites for Using Ansible for Oracle Cloud Infrastructure

- You must have an Oracle Cloud Infrastructure account.
- Create a user in that account, in a security group with a policy that grants necessary permissions for working with resources in the account compartments.
- You must have the necessary credentials and OCID information.

Installing the Oracle Cloud Infrastructure Python SDK

1. Download and install the Python SDK by following instructions in the topic, Python SDK. For additional guidance, see Downloading and Installing the SDK.
2. After installing the Python SDK, you must configure it using instructions in the topic Configuring the SDK.

Installing and Configuring Ansible

- To install Ansible, follow the instructions provided in the Ansible Installation Guide.
- For guidance configuring Ansible, see Configuring Ansible.

Installing Oracle Cloud Infrastructure Ansible Modules

You can install the Ansible modules for Oracle Cloud Infrastructure using the installation script at the Oracle GitHub Ansible repository. To install the Ansible modules using the install script, run the following commands:

1. `$ git clone https://github.com/oracle/oci-ansible-modules.git`
2. `$ cd oci-ansible-modules`
3. Run one of the following commands:
    a. If Ansible is installed as a user: `$ ./install.py`
    b. If Ansible is installed as root: `$ sudo ./install.py`

Sample Ansible Modules

Sample modules are available in the Oracle Cloud Infrastructure Ansible Module GitHub project. The samples library is updated regularly with the addition of new samples. You can access the samples at https://github.com/oracle/oci-ansible-modules.

Writing a Sample Playbook

You can now write a sample playbook that uses Ansible modules. Following is an example playbook (named `list_buckets.yml`) that uses the `oci_bucket_facts` module to fetch all of the facts pertaining to all of the buckets in your compartment.

```
---
- name : List summary of existing buckets in OCI object storage
  connection: local
  hosts: localhost
  tasks:
    - name: List bucket facts
      oci_bucket_facts:
         namespace_name: '<yournamespace>'
         compartment_id: '<yourcompartmentocid>'
      register: result
    - name: Dump result
      debug:
        msg: '{{result}}'
```

Executing the Playbook

Execute the Ansible playbook using Python by invoking this command:

```
$ ansible-playbook list_buckets.yml
```

How to Obtain Module Documentation

To obtain access to detailed information about using Ansible modules in the CLI, including documentation of a module's configurable options, samples, return values, and so forth, use the `ansible-doc` command on the module's name. For example, to get the documentation for the `oci_bucket_facts` module, execute the following command:

```
$ ansible-doc oci_bucket_facts
```

Documentation of the Oracle Cloud Infrastructure Ansible modules is also available in the Cloud Modules page of the Oracle Cloud Infrastructure Ansible Modules site.

Configuring Authentication

When creating and configuring Oracle Cloud Infrastructure resources, Ansible modules use authentication information that is outlined in SDK and CLI Configuration File.



Warning

IAM credentials that are referenced in Oracle Cloud Infrastructure SDK configuration files grant access to Oracle Cloud Infrastructure resources. Because of this, it is important to secure the credentials to prevent unauthorized access to these resources. To secure the credentials on the controller node where your Ansible playbooks run, follow guidelines outlined in the document Securing IAM (see section entitled "IAM Credentials").

Ansible modules permit you to override authentication information specified in the SDK configuration file by using module options and environment variables. Documentation for this is provided internally, as described in the preceding section, **How to Obtain Module Documentation**. However, using environment variables and Ansible module options to override authentication information must be avoided in production scenarios.

We recommend using Oracle Cloud Infrastructure SDK configuration files to specify authentication information. Use the "profiles" feature in the SDK configuration file to support different users. When distributing roles that use Ansible modules, ensure that no IAM credentials are included with the roles.

For More Information

- Sample Ansible Playbooks
- Get Started with Ansible
- How Ansible Works
- Ansible for DevOps

Using the Dynamic Inventory Script

Ansible tracks configuration resources by preserving lists, called inventory lists, as simple list files (also sometimes called a *hostfile*). These inventory files can be either simple static lists,

or they can be dynamic lists that automatically update when inventory resources are added, deleted, or moved. For more information about Ansible inventory files, see Working with Inventory. See also, Working with Dynamic Inventory.

When using Ansible to work with hosts that you have provisioned in Oracle Cloud Infrastructure, static inventory lists can cause problems because Compute instances are added and deleted over time. They can also be affected by external tools such as Terraform, or by the Oracle Cloud Infrastructure SDKs.

Having up-to-date and accurate inventory lists is essential for running Ansible playbooks. Oracle Cloud Infrastructure provides you with a script that you can download and run to ensure that your instance inventory list is always up-to-date. The script ensures that you always have the current set of Oracle Cloud Infrastructure compute instances available to your playbooks.

Download and Configure the Dynamic Inventory Script

Download the dynamic inventory script (`oci_inventory.py`) and the default configuration file (`oci_inventory.ini`) from the following:

- https://github.com/oracle/oci-ansible-modules/blob/master/inventory-script/oci_inventory.py
- https://github.com/oracle/oci-ansible-modules/blob/master/inventory-script/oci_inventory.ini



Important

Before you can use the script, ensure that you have a valid Oracle Cloud Infrastructure configuration. For guidance configuring `~/.oci/config`, see SDK and CLI Configuration File.

Script and Configuration Details

The Python script, `oci_inventory.py`, uses the Oracle Cloud Infrastructure Python SDK to query compute instances in your Oracle Cloud Infrastructure tenancy and then uses this

information to create a dynamic inventory that you can use with your Ansible playbooks. Use arguments in the Python script to control the configuration profile and the tenancy compartment that you query against.

You can use the configuration file, `oci_inventory.ini`, to control how inventory details are cached, and to control which Oracle Cloud Infrastructure profile to use. This file allows you to modify host names, and to control how hosts are named in the inventory list that is generated.

Arguments and Environment Variables

For a complete list of command-line arguments and environment variables that the script accepts, see Dynamic Inventory Script.

Order of Precedence

Following is the order of precedence for configurations that are used by the dynamic inventory script:

1. Command-line arguments.
2. Environment variables.
3. Configuration settings in the selected `profile` in your Oracle Cloud Infrastructure configuration file.



Note

The default configuration file used by the inventory script is `./oci_inventory.ini`. The Oracle Cloud Infrastructure SDK configuration file, however, defaults to `~/.oci/config`. The script reads the `DEFAULT` profile from the `config` file if no profile name is specified.



Important

By default, the inventory is generated for all the compartments in the tenancy. You must have `COMPARTMENT_INSPECT` permission on the root compartment for this script to be able to access all compartments. However, when `compartment_ocid` is specified, the inventory is generated for only the specific compartment, so you only need `COMPARTMENT_INSPECT` permission on the specified compartment.

The inventory list that is generated by the dynamic inventory script is grouped using the following attributes:

- The region in which the compute instance resides.
- The name of the compartment the compute instance belongs to.
- The Availability Domain the compute instance is in.
- The vcn_id of the vcn the compute instance is in.
- The subnet_id of the subnet the compute instance is in.
- The security_list_ids of the subnet the compute instance is in.
- The image_id of the image used to launch the compute instance.
- Shape of the compute instance.
- The instance's free-form tags, with the group name set to tag_<tag_name>=<tag_value>.
- The instance's defined tags, with the group name set to <tag_namespace>#<tag_name>=<tag_value>.
- Oracle Cloud Infrastructure compute instance metadata (key-value pairs), with the group name set to <metadata-key>=<metadata-value>.
- Oracle Cloud Infrastructure compute instance extended metadata (key-value pairs), with the group name set to <metadata-key>=<metadata-value>.



Important

By default, all non-alphanumeric characters in group names and host names are replaced with an underscore (_) when the inventory is generated *except* hash (#), equals (=), period (.) and dash (-) . This allows you to use these names as Ansible group names. To disable this default substitution, set `sanitize_names` to `False` in the dynamic inventory settings file, whose default location is `./oci_inventory.ini`). To also replace the dash (-) when `sanitize_names` is `True`, set `replace_dash_in_names` to `True` in the settings file.

How to Use Dynamic Inventory

Using Dynamic Inventory During Playbook Execution

To use your dynamic inventory, first ensure that you have a correct Oracle Cloud Infrastructure SDK configuration file. Optionally, you can also have an `oci_inventory.ini` file.

Invoke `ansible-playbook` using the following command:

```
$ ansible-playbook -i <path-to-inventory-file>/oci_inventory.py <your-playbook-using-the-generated-inventory>
```

Alternatively, use the `ANSIBLE_HOSTS` environment variable by using the following command:

```
$ ANSIBLE_HOSTS=<path-to-inventory-file>/oci_inventory.py ansible-playbook <your-playbook-using-the-generated-inventory>
```

Disabling the Cache and Fetching the Latest Inventory List

If you are running the dynamic inventory script in a stand-alone manner, you can ignore the cached inventory list and fetch the most current inventory list by using the `--refresh` (`-r`) argument, as shown in the following example:

```
$ \<path-to-inventory-file\>/oci_inventory.py --refresh
```

If instead you use the inventory script during an Ansible playbook invocation, set the `OCI_CACHE_MAX_AGE` environment variable to `0` (zero) to ignore the cached list, and fetch the latest compute instance, as shown in the following example:

```
$ OCI_CACHE_MAX_AGE=0 ansible-playbook -i <path-to-inventory-file>/oci_inventory.py <your-playbook-using-the-generated-inventory>
```

DEBUGGING THE INVENTORY LIST

To examine the inventory list, run the dynamic inventory script using the `--list` argument, as shown here:

```
$ \<path-to-inventory-file\>/oci_inventory.py --list
```

If you wish to print additional debug information to `STDERR`, use the `--debug` argument, as shown:

```
$ \<path-to-inventory-file\>/oci_inventory.py --debug
```

RETRIEVE INFORMATION ABOUT A HOST

You can configure the inventory script to provide information about a specified host by using the `--host` argument and providing the host's IP address, as shown:

```
$ \<path-to-inventory-file\>/oci_inventory.py --host <host IP address>
```

The command returns the following set of variables and values:

```
{
  "availability_domain": "IwGV:US-ASHBURN-AD-1",
  "compartment_id": "ocid1.compartment.oc1..<xxxxxEXAMPLExxxxx>",
  "defined_tags": {},
  "display_name": "ansible-test-instance",
  "extended_metadata": {},
  "freeform_tags": {},
  "id": "ocid1.instance.oc1.iad.<xxxxxEXAMPLExxxxx>",
  "image_id": "ocid1.image.oc1.iad.<xxxxxEXAMPLExxxxx>",
  "ipxe_script": null,
  "launch_mode": "CUSTOM",
  "launch_options": {
     "boot_volume_type": "ISCSI",
     "firmware": "UEFI_64",
     "network_type": "VFIO",
     "remote_data_volume_type": "ISCSI"
```

```
  },
  "lifecycle_state": "AVAILABLE",
  "metadata": {
     "baz": "quux",
     "foo": "bar"
  },
  "region": "iad",
  "shape": "VM.Standard1.1",
  "source_details": {
    "image_id": "ocid1.image.oc1.iad.<xxxxxEXAMPLExxxxx>",
  "source_type": "image"
  },
  "time_created": "2018-01-16T12:13:35.336000+00:00"
}
```

TROUBLESHOOTING THE DYNAMIC INVENTORY SCRIPT

If the inventory list generated by the inventory script does not include all of the compute instances in your tenancy, review the following information.

USER PERMISSIONS

Ensure that the user OCID (specified using either the `OCI_USER` environment variable, or the `profile` section in your SDK configuration file) has the policy permissions to list the compute instances. To see a list of permissions for API operations, see Details for the Core Services.

The inventory script makes API calls for the following operations:

- ListCompartments
- ListVNICAttachments
- GetSubnet
- GetVCN
- GetVNIC
- GetInstance

HOSTNAME FORMAT

The default for `OCI_HOSTNAME_FORMAT` is `public_ip`. The dynamic inventory generated using `OCI_HOSTNAME_FORMAT` set to `public_ip` contains only compute instances that have a public

IP address. This can be useful in cases where the Ansible controller node is outside the VCN, since Ansible can only reach instances that have public IP addresses.

However, if running Ansible in a compute instance within your VCN, and it has access to all of the subnets within your VCN, including compute instances that have private IP addresses, you must set the `OCI_HOSTNAME_FORMAT` to `private_ip` to list compute instances using their private IP addresses.

Sample Ansible Playbooks

Provided here is a catalog of sample Ansible playbooks for Oracle Cloud Infrastructure that illustrate how to carry out common infrastructure provisioning and configuration tasks. The samples are organized in groups associated with Oracle Cloud Infrastructure services:

- Block Volume
- Compute
- Container Engine for Kubernetes
- Database
- File Storage
- IAM
- Load Balancing
- Object Storage
- Deployment Solution (MongoDB)

You find a brief description of each playbook in the sections that follow, along with links to each sample on the Oracle GitHub repository. Begin by reviewing the Readme.md file that you find in each playbook's root directory.

BLOCK VOLUME

ATTACH A BLOCK VOLUME TO A COMPUTE INSTANCE

This sample playbook shows how to attach a block volume to a compute instance using the iSCSI volume attachment type, and then connect it to the compute instance using `iscsiadm`. The sample shows how to do the following:

- Generate a temporary, host-specific SSH key pair.
- Specify the public key from the key pair for connecting to the instance, and then launch the instance.
- Create a new Block Volume for the instance, attach the volume to the instance, and specify `iSCSI` as the volume attachment type.
- Connect to and then mount the volume from the compute instance by executing `iscsiadm` commands over SSH using an Ansible module.

Go to the sample on Oracle GitHub.

COMPUTE

LAUNCH A COMPUTE INSTANCE

This sample playbook shows how to launch a public Compute instance and then access the instance from an Ansible module over an SSH connection. The sample illustrates how to do the following:

- Generate a temporary, host-specific SSH key pair.
- Specify the public key from the key pair for connecting to the instance, and then launch the instance.
- Connect to the newly launched instance using SSH.

Go to the sample on Oracle GitHub.

USE NAT TO ENABLE INTERNET ACCESS FROM A COMPUTE INSTANCE

This sample playbook shows how to enable internet access from a Compute instance in a private subnet. The example uses a network address translation (NAT) instance in a public

subnet through an Ansible module. The sample illustrates how to do the following:



Note

For guidance setting up the network topology to support a NAT instance, see the white paper NAT Instance Configuration: Enabling Internet Access for Private Subnets. See also Tutorial: Automatically Set Up a NAT Instance in Oracle Cloud Infrastructure with Terraform.

- Set up the applicable network topology, including creating the VCN, internet gateway, public and private subnets, and required security lists and route rules.
- Provision a NAT instance in the public subnet, and a private instance in the private subnet.
- Enable outbound internet access for the private instance through the NAT instance on the public subnet.

Go to the sample on Oracle GitHub.

ENABLE INTERNET ACCESS FROM A COMPUTE INSTANCE USING THE ORACLE CLOUD INFRASTRUCTURE NAT GATEWAY

This sample is similar to the previous sample except that while the previous sample configures a compute instance to operate as a NAT Gateway, the present sample employes the Oracle Cloud Infrastructure NAT Gateway service.



Note

For more information about the Oracle Cloud Infrastructure NAT Gateway service, see NAT Gateway. For a blog post discussing how to use the Oracle Cloud Infrastructure NAT Gateway, see Access Resources on the Public Internet Through an Oracle Cloud Infrastructure NAT Gateway.

The sample show how to complete the following:

- Set up the VCN, the NAT Gateway, the Internet Gateway, the public and private subnets, and the necessary security lists and route rules.
- Provision a bastion instance in the public subnet and a private instance in the private subnet.

Once set up, the private instance will have outbound Internet access through the Oracle Cloud Infrastructure NAT Gateway, and will be accessible using SSH from the bastion instance.

Go to the sample on Oracle GitHub

CREATE AN INSTANCE POOL

This sample shows how to manage your Compute instances using resources such as instance configurations and instance pools that are provided using Oracle Cloud Infrastructure Ansible modules. Instance pools help you create and provision multiple Compute instances within the same region based on a single instance configuration.

The sample illustrated completing the following tasks:

- Generate a temporary, host-specific SSH key pair.
- From the SSH key pair, specify the public key for connecting to the instance during launch.
- Create an instance configuration that defines settings for creating a Compute instance as part of the instance pool. The configuration provides details such as base image, shape, and metadata.
- Demonstrates how the Compute instances based on the instance configuration can be launched using instance pools.
- Connect to one of the Compute instances using SSH.

Go to the sample on Oracle GitHub

CREATE INSTANCE CONSOLE CONNECTIONS AND CAPTURE CONSOLE HISTORY

This sample shows you can create VNC and serial console connections to a Compute instance, and how you can fetch and capture the serial Console data from the instance. For more

information about Console connections, see Instance Console Connections.

This sample illustrates completing the following tasks:

- Generate a temporary SSH key pair for the serial Console connection.
- Create an instance Console connection for a Compute instance.
- Capture serial Console data for a Compute instance, and then save the data to a local machine so you can troubleshoot and debug issues.

Go to the sample on Oracle GitHub

ACCESS OBJECT STORAGE FROM A PRIVATE INSTANCE USING SERVICE GATEWAY

This sample playbook hows how you can enable private access to anObject Storage from a Compute instance using a service gateway.



Note

For more information about service gateways, see Access to Oracle Services: Service Gateway. To read a blog post discussing how to connect Compute instances using the service gateway, see Connect Private Instances with Oracle Services Through an Oracle Cloud Infrastructure Service Gateway.

The sample shows how to complete the following tasks:

- Set up a user, group, and policies required for managing buckets.
- Create and upload the required API keys to the user.
- Set up the VCN, the NAT gateway, the Internet gateway, the public and private subnets, as well as the required security lists and route tables. Note that a bastion instance is provisioned in the public subnet, and a private instance is provisioned in the private subnet.
- Provision a Compute instance in the private subnet,

- Install the Oracle Cloud Infrastructure command line interface (CLI) and configure the CLI using the cloud init script.
- Disable the NAT gateway to restrict public access to the private instance.
- Create a bucket from the private instance using the Oracle Cloud Infrastructure CLI, then verify that the bucket is created.

Following this setup, the private instnace will have private access to Object Storage.

Go to the sample on Oracle GitHub

CONTAINER ENGINE FOR KUBERNETES

CREATE A CLUSTER USING CONTAINER ENGINE FOR KUBERNETES

This sample playbook uses Container Engine for Kubernetes (OKE) to create a cluster and deploys a sample application on the cluster. This sample complements an existing example, Creating a Cluster with Oracle Cloud Infrastructure Container Engine for Kubernetes.

This sample illustrates how to do the following:

- Creates and configures a VCN and related resources required for setting up an OKE cluster.
- Creates a cluster.
- Creates a node pool.
- Downloads the kubeconfig file for the cluster.
- Deploys a sample application on the cluster.
- Verifies a successful deployment.

Go to the sample on Oracle GitHub.

DATABASE

BARE METAL/VM DATABASE PROVISIONING

This sample playbook shows how to retrieve the public and private IP addresses of a database system node so that you can access it through an Ansible module. The sample illustrates how

to do the following:

- Collect database node VNIC information for a specified database.
- Extract public and private IP addresses of the database node from the VNIC.

Go to the sample on Oracle GitHub.

AUTONOMOUS DATA WAREHOUSE

This sample playbook shows how to create an Autonomous Data Warehouse and manage its lifecycle. The sample shows how to do the following:

- Set up an Autonomous Data Warehouse.
- List all of the Autonomous Data Warehouse instances available in a compartment, filtered by display name.
- Get the "facts" for a specified Autonomous Data Warehouse.
- Stop and start an Autonomous Data Warehouse instance.
- Delete an Autonomous Data Warehouse instance.

Go to the sample on Oracle GitHub.

AUTONOMOUS TRANSACTION PROCESSING

This sample playbook shows how to create an Autonomous Transaction Processing database and manage its lifecycle. The sample shows how to do the following:

- Set up an Autonomous Transaction Processing database instance.
- List all of the Autonomous Transaction Processing instances in a compartment, filtered by display name.
- Get the "facts" for a specified Autonomous Transaction Processing instance.
- Delete an Autonomous Transaction Processing database instance.

Go to the sample on Oracle GitHub.

File Storage

Create and Mount a File System

The sample shows how to create a file system that you can access through an Oracle Cloud Infrastructure Compute instance using Ansible cloud modules. The sample illustrates completing the following tasks:

- Creates network dependencies like VCNs, subnets, and so forth, as well as a security list that is configured as required by the File Systems service.
- Generates the certificates required by the Compute instances.
- Demonstrates how to create File Storage service components, such as mount targets, file systems, exports, and snapshots.
- Demonstrates how to mount the file system using a Compute instance, and how to then access the file system content from a different Compute instance.

Go to the sample on Oracle GitHub

Multiple File Systems with Mount Targets

This sample shows how one you can export one file system using two different export paths located on two different mount targets. It also shows how a single mount target can export paths from two different file systems. The sample illustrates completing the following tasks:

- Creates network dependencies like VCNs, subnets, and so forth, as well as a security list that is configured as required by the File Systems service.
- Generates the certificates required by the Compute instances.
- Demonstrates how to create File Storage service components, such as mount targets, file systems, exports, and snapshots.
- Demonstrates how one file system can be exported onto two different mount targets.
- Demonstrates how a single mount target can export paths from two different file systems.
- Demonstrates how to mount the file system using an Oracle Cloud Infrastructure Compute instance.

Go to the sample on Oracle GitHub

IAM

Use Ansible Modules to Perform IAM Tasks

This sample shows how to perform basic identity and access management (IAM) tasks using Ansible modules. The sample also shows how to execute an Ansible playbook or execute individual tasks as a different user. The sample illustrates how to do the following:

- Create groups (**ObjectReaders** and **ObjectWriters**).
- Create an IAM policy the enables the following:
  - **ObjectReaders** to list and read buckets and objects.
  - **ObjectWriters** to create, update, list, and read buckets and objects in a specified compartment.
  - Assign the policy to groups.
- Create users (**alice** and **bob**) and then do the following:
  - Add **alice** to the **ObjectWriters** group.
  - Add **bob** to the **ObjectReaders** group.
  - Run as **alice** to create a bucket, then upload objects to the bucket.
  - Run as **bob** to list all objects in a bucket.

Go to the sample on Oracle GitHub.

Load Balancing

This sample playbook shows how to create a public load balancer using an Ansible module. The sample illustrates the following:

- Generating network-related artifacts, such as subnets and VCNs, for example.
- Generating the required certificates for the load balancer.
- Using an Ansible playbook to create a public load balancer.

Go to the sample on Oracle GitHub.

OBJECT STORAGE

LIST OBJECTS AND BUCKETS

This sample playbook shows how to list all objects and buckets in a namespace.

Go to the sample on Oracle GitHub.

DELETE OBJECTS

This sample playbook shows how to delete objects created within a specified range of days from the specified the buckets. You can also modify the sample so it deletes objects older than a specified number of days, which helps you prune old or unwanted objects that are stored in the service.

Go to the sample on Oracle GitHub.

MONGODB DEPLOYMENT

USE ANSIBLE MODULES TO DEPLOY A MONGODB DATABASE

This sample playbook shows how to deploy a MongoDB database in the securely in the cloud using Ansible modules. The sample implements security measures using the Castle strategy ("defense in depth"), which is discussed in the article Secure MongoDB on Oracle Bare Metal Cloud Services.

Go to the sample on Oracle GitHub.

NETWORKING

USE ANSIBLE MODULES TO PROVISION A VCN

This sample playbook shows how to provision a virtual cloud network (VCN) with two private subnets in different availability domains, and an IPSec VPN.The sample provisions infrastructure resources that are illustrated in the knowledge base article Scenario B: Private Subnet with a VPN. The sample provisions the following resources:

- A VCN.
- Two private subnets.

- A dynamic routing gateway (DRG).
- Customer premises equipment (CPE).
- An IPSec connection between the DRG and the CPE, and retrieves IPSec configuration information and status.

Go to the sample on Oracle GitHub.

Chef Knife Plug-in

This topic provides information about installing, configuring, and using the Chef Knife Plug-in for Oracle Cloud Infrastructure.

- **Licensing:** This provider and sample is licensed under the Mozilla Public License 2.0; third-party content is separately licensed as described in the code.
- **Download:** GitHub
- **Documentation:** README

Contributions

Got a fix for a bug, or a new feature you'd like to contribute? The Chef Knife Plug-in for Oracle Cloud Infrastructure is open source and accepting pull requests on GitHub.

Notifications

To be notified when a new version of the Chef Knife Plug-in for Oracle Cloud Infrastructure is released, subscribe to the Atom feed.

Questions or Feedback

- GitHub: To file bugs and feature requests only.
- Stack Overflow: Please use the oracle-cloud-infrastructure tag in your post.

- Developer Tools section of the Oracle Cloud forums
- My Oracle Support

Compute Jenkins Plug-in

This topic provides information about installing, configuring, and using the Compute Jenkins plug-in for Oracle Cloud Infrastructure services.

- **Licensing:** The Oracle Cloud Infrastructure Compute Jenkins plug-in is dual-licensed under the Universal Permissive License (UPL) and the Apache License 2.0; third-party content is separately licensed as described in the code.
- **Download:** GitHub
- **Documentation:** Oracle Cloud Infrastructure Compute Plug-in - Jenkins Wiki

Contributions

Got a fix for a bug, or a new feature you'd like to contribute? The Oracle Cloud Infrastructure Compute Jenkins plug-in is open source and accepting pull requests on GitHub.

Notifications

To be notified when a new version of the Oracle Cloud Infrastructure Compute Jenkins plug-in is released, subscribe to the Atom feed.

Questions or Feedback

- GitHub: To file bugs and feature requests only.
- Stack Overflow: Please use the oracle-cloud-infrastructure tag in your post.
- Developer Tools section of the Oracle Cloud forums
- My Oracle Support

Grafana Plug-in

This topic provides instructions for installing, configuring, and using Oracle Cloud Infrastructure Data Source for Grafana, otherwise referenced as the Grafana Plug-in.

Grafana Plug-in Overview

Grafana is an open-source visualization and alerting tool that you can use for analytics and monitoring of time-series data (metrics). While metrics from Oracle Cloud Infrastructure Monitoring are visible in metrics charts through the Console, you can use Oracle Cloud Infrastructure Data Source for Grafana ("the Grafana Plug-in") to view metrics from resources across providers on a single Grafana dashboard.

Prerequisites for Using the Grafana Plug-in

- An Oracle Cloud Infrastructure account.
- A user in that account, in a security group with an IAM policy that grants necessary permissions for working with resources in the account compartments. The policy must give you access to the metric namespaces emitting metrics (such as Compute) as well as the related resources (such as a set of Compute instances). If you try to perform an action and get a message that you don’t have permission or are unauthorized, confirm with your administrator the type of access you've been granted and which compartment you should work in. Administrators: For common policies that give groups access to metrics, see Let users view metric definitions in a compartment and Restrict user access to a specific metric namespace. To authorize resources, such as instances, to make API calls, add the resources to a dynamic group through its matching rules, and then create a policy that allows that dynamic group access to metrics. To allow access across compartments, create the policy in the tenancy. Because of the concept of policy inheritance, instances in the indicated dynamic group can then access metrics in any compartment. To reduce the scope of access to a particular compartment, specify that compartment instead of the tenancy. See Let instances make API calls to access monitoring metrics in the tenancy.

- Compute instances: To emit metrics, Compute instances must be monitoring-enabled. OracleCloudAgent software installation may also be required. For more information, see Enabling Monitoring for Compute Instances.
- Required keys and Oracle Cloud Infrastructure IDs (OCIDs). For guidance, see "Required Keys and OCIDs" in the *Oracle Cloud Infrastructure User Guide*.

Download and Install the Grafana Plug-in

This section provides instructions for downloading and installing the Grafana Plug-in, either locally (Linux or Mac) or by Terraform script. Use Terraform script to provision a Compute instance that runs Grafana in Oracle Cloud Infrastructure.

Authentication for metric access depends on where Grafana is running. If you are running Grafana on a local machine outside Oracle Cloud Infrastructure, you must call the Monitoring API using the Command Line Interface (CLI). If you are running Grafana on the new Compute instance created by the Terraform script, use the script-generated dynamic group. If you are running Grafana on an Oracle Cloud Infrastructure Compute instance that you created, then you can add the instance to a dynamic group by configuring a matching rule. In all scenarios, you have the option of calling the API using the CLI.

Local: Download and Install the Plug-in

Local Installation Prerequisites

Grafana installation. Version 3.0 or later required.

To install the Grafana Plug-in (Oracle Cloud Infrastructure Data Source for Grafana), see https://grafana.com/plugins/oci-datasource/installation. After installation, you're ready to configure the Grafana Plug-in.

Terraform Script: Create a Grafana environment

This section describes how to use Terraform scripts to create a Grafana environment on your virtual machine.

The Terraform scripts create a dynamic group using the name you specify, configure an IAM policy named "grafana_policy," provision a Compute instance named "TFInstance0" (in

the compartment and VCN that you specify), and download and install both Grafana and the Grafana Plug-in onto the instance.

Terraform Script Prerequisites

Before running the Terraform scripts, make sure you have the following:

- Terraform and Oracle Cloud Infrastructure Terraform Provider must both be installed on your development machine. See Getting Started with the Terraform Provider.
- A Virtual Cloud Network (VCN) with access to the Internet.
  For reference, see the Linux Tutorial for creating a VCN.
- Values for the following variables. You can pass the values to your .bashrc or .bash_profile, or note them somewhere for entering when needed. These variables are required for creating the Grafana environment and using the Grafana Plug-in.

Required variables

```
variable "tenancy_ocid" {}
variable "user_ocid" {}
variable "fingerprint" {}
variable "private_key_path" {}
variable "region" {}
variable "compartment_ocid" {}
variable "ssh_public_key" {}
variable "ssh_private_key" {}
variable "subnet_id" {}
variable "availability_domain" {}
variable "dynamic_group_name" {}
```



Note

For help on getting the tenancy OCID and other values needed for these variables, see Required Keys and OCIDs.

To download and run the Terraform scripts

Make sure you satisfy the prerequisites before running the scripts.

1. Download the Terraform scripts using the following command.
   `wget https://objectstorage.us-ashburn-1.oraclecloud.com/n/oracle-cloudnative/b/GrafanaTerraform/o/terraform_grafana.tar && tar -xvf terraform_grafana.tar`
2. Change to the terraform_grafana directory using the following command.
   `cd /terraform_grafana`
3. Initialize a new Terraform configuration using the following command.
   `terraform init`
   Initialization creates the files `terraform.tfstate` and `terraform.tfstate.backup`.
4. Generate an execution plan using the following command.
   `terraform plan`
   Generating a plan can help you validate your Terraform script before you apply it to your environment.
5. Run the scripts using the following command, using the same availability domain and dynamic group variables passed in with `terraform plan`.
   `terraform apply`
   On completion, the "Apply complete!" message appears. You're now ready to configure the Grafana Plug-in.

Configure the Grafana Plug-in

This section describes how to add the Grafana Plug-in and set up a dashboard.

To configure the Grafana Plug-in

Local installation



Note

When installed locally, the Grafana plug-in accesses metrics using calls to the Monitoring API.

1. Set up the CLI for accessing Oracle Cloud Infrastructure APIs.
   You'll need to access the Monitoring API for authentication.
2. Navigate to the Grafana homepage at the following URL.
   http://localhost:3000
3. Add the Grafana Plug-In (Oracle Cloud Infrastructure Data Source for Grafana).
   In addition to the steps below, see the Grafana instructions for adding data sources at http://docs.grafana.org/guides/getting_started/.
   a. In Grafana, on the Home Dashboard, click the gear icon on the left.
   b. Click **Add data source**.
   c. In the Filter text box, type: `oracle-oci-datasource`
   d. In the filtered list, select **oracle-oci-datasource**.
   e. In the Settings page, fill in your **Tenancy OCID**, **Default Region**, and **Environment**. For **Environment** choose **local**.

Environment options

local is for Grafana deployments outside Oracle Cloud Infrastructure.

OCI Instance is for Grafana deployments on Oracle Cloud Infrastructure resources.

The data source is now added, enabling you to set up a dashboard showing metrics from Oracle Cloud Infrastructure.

4. Set up a dashboard of the type "Graph."
5. (Optional) To confirm access to metrics from Oracle Cloud Infrastructure, update your dashboard query ("Metrics") with a specific region, compartment, namespace, metric. You can also add one or more dimensions.
   Congratulations. You can now view your Oracle Cloud Infrastructure metrics in Grafana!

Terraform environment



Note

When installed by Terraform script, the Grafana plug-in accesses metrics using the script-generated dynamic group.

1. Connect to the new instance "TFInstance0."
   For instructions, see Connecting to an Instance.
2. Navigate to the Grafana homepage by running the following command, where *[Instance Public IP]* is the IP address of your new instance "TFInstance0."
   `ssh opc@[Instance Public IP] -L 3000:localhost:3000`



Note

You can get the address from the list of instances in the Console. Click **Compute**, choose your **Compartment**, and then find your instance in the list. Alternatively, you can use the Core Services API`ListVnicAttachments` and `GetVnic` operations.

3. Add the Grafana Plug-In (Oracle Cloud Infrastructure Data Source for Grafana).
   In addition to the steps below, see the Grafana instructions for adding data sources at http://docs.grafana.org/guides/getting_started/.
   a. In Grafana, on the Home Dashboard, click the gear icon on the left.
   b. Click **Add data source**.
   c. In the Filter text box, type: `oracle-oci-datasource`
   d. In the filtered list, select **oracle-oci-datasource**.
   e. In the Settings page, fill in your **Tenancy OCID**, **Default Region**, and **Environment**. For **Environment** choose **OCI Instance**.

      Environment options

      local is for Grafana deployments outside Oracle Cloud Infrastructure.
      OCI Instance is for Grafana deployments on Oracle Cloud Infrastructure resources.

      The data source is now added, enabling you to set up a dashboard showing metrics from Oracle Cloud Infrastructure.
4. Set up a dashboard of the type "Graph."
5. (Optional) To confirm access to metrics from Oracle Cloud Infrastructure, update your dashboard query ("Metrics") with a specific region, compartment, namespace, metric.

You can also add one or more dimensions.

Congratulations. You can now view your Oracle Cloud Infrastructure metrics in Grafana!

Troubleshoot the Plug-In

If the dashboard query ("Metrics") fails to populate with options, or if you have other issues accessing metrics, then the IAM policy used to access metrics may be malformed, or it may not include all required matching rules for your dynamic group (Terraform script).

To resolve this issue, do the following.

- Review your IAM policy to ensure that it matches prerequisites.
- For Terraform, consider adding the following matching rule to your dynamic group: `matching_rule = "ANY {instance.compartment.id = '${var.compartment_ocid}'}"`
- If you updated your IAM policy, then restart the Grafana server and refresh the Grafana homepage.

Terraform: Remove the environment

If you ran the Terraform script to create a Grafana environment on your virtual machine, you may want to remove the environment after your work is done. Follow the instructions in this section to do so.

To remove your Grafana environment

Run the following command, using the same availability domain and dynamic group variables passed in with `terraform apply`.

```
terraform destroy
```

Tools Configuration

This general reference shows how to configure the SDKs and other developer tools to integrate with Oracle Cloud Infrastructure services.

- Required Keys and OCIDs - Details on identity and access management
- SDK and CLI Configuration File - Methods for providing configuration information when using the SDKs or CLI.

Required Keys and OCIDs

Whether you're using an Oracle client (see Software Development Kits and Command Line Interface) or a client you built yourself, you need to do the following:

1. Create a user in IAM for the person or system who will be calling the API, and put that user in at least one IAM group with any desired permissions. See "Adding Users" in the *Oracle Cloud Infrastructure Getting Started Guide*. You can skip this if the user exists already.
2. Get these items:
    - RSA key pair **in PEM format** (minimum 2048 bits). See How to Generate an API Signing Key.
    - Fingerprint of the public key. See How to Get the Key's Fingerprint.
    - Tenancy's OCID and user's OCID. See Where to Get the Tenancy's OCID and User's OCID.
3. Upload the public key from the key pair in the Console. See How to Upload the Public Key.
4. If you're using one of the Oracle SDKs or tools, supply the required credentials listed above in either a configuration file or a config object in the code. See SDK and CLI Configuration File. If you're instead building your own client, see Request Signatures.



Important

This key pair is **not** the SSH key that you use to access compute instances. See Security Credentials.

Both the private key and public key must be in PEM format (not SSH-RSA format). The public key in PEM format looks something like this:

```
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQE...
...
-----END PUBLIC KEY-----
```

How to Generate an API Signing Key

You can use the following OpenSSL commands to generate the key pair in the required PEM format. If you're using Windows, you'll need to install Git Bash for Windows and run the commands with that tool.

1. If you haven't already, create a `.oci` directory to store the credentials:

   ```
   mkdir ~/.oci
   ```

2. Generate the private key with one of the following commands.

   - Recommended: To generate the key, encrypted with a passphrase you provide when prompted:

     ```
     openssl genrsa -out ~/.oci/oci_api_key.pem -aes128 2048
     ```

     Note: For Windows, you may need to insert `-passout stdin` to be prompted for a passphrase. The prompt will just be the blinking cursor, with no text.

     ```
     openssl genrsa -out ~/.oci/oci_api_key.pem -aes128 -passout stdin 2048
     ```

   - To generate the key with no passphrase:

     ```
     openssl genrsa -out ~/.oci/oci_api_key.pem 2048
     ```

3. Ensure that only you can read the private key file:

```
chmod go-rwx ~/.oci/oci_api_key.pem
```

4. Generate the public key:

```
openssl rsa -pubout -in ~/.oci/oci_api_key.pem -out ~/.oci/oci_api_key_public.pem
```

Note: For Windows, if you generated the private key with a passphrase, you may need to insert `-passin stdin` to be prompted for the passphrase. The prompt will just be the blinking cursor, with no text.

```
openssl rsa -pubout -in ~/.oci/oci_api_key.pem -out ~/.oci/oci_api_key_public.pem -passin stdin
```

5. Copy the contents of the public key to the clipboard using pbcopy, xclip or a similar tool (you'll need to paste the value into the Console later). For example:

```
cat ~/.oci/oci_api_key_public.pem | pbcopy
```

Your API requests will be signed with your private key, and Oracle will use the public key to verify the authenticity of the request. You must upload the public key to IAM (instructions below).

How to Get the Key's Fingerprint

You can get the key's fingerprint with the following OpenSSL command. If you're using Windows, you'll need to install Git Bash for Windows and run the command with that tool.

```
openssl rsa -pubout -outform DER -in ~/.oci/oci_api_key.pem | openssl md5 -c
```

When you upload the public key in the Console, the fingerprint is also automatically displayed there. It looks something like this: 12:34:56:78:90:ab:cd:ef:12:34:56:78:90:ab:cd:ef

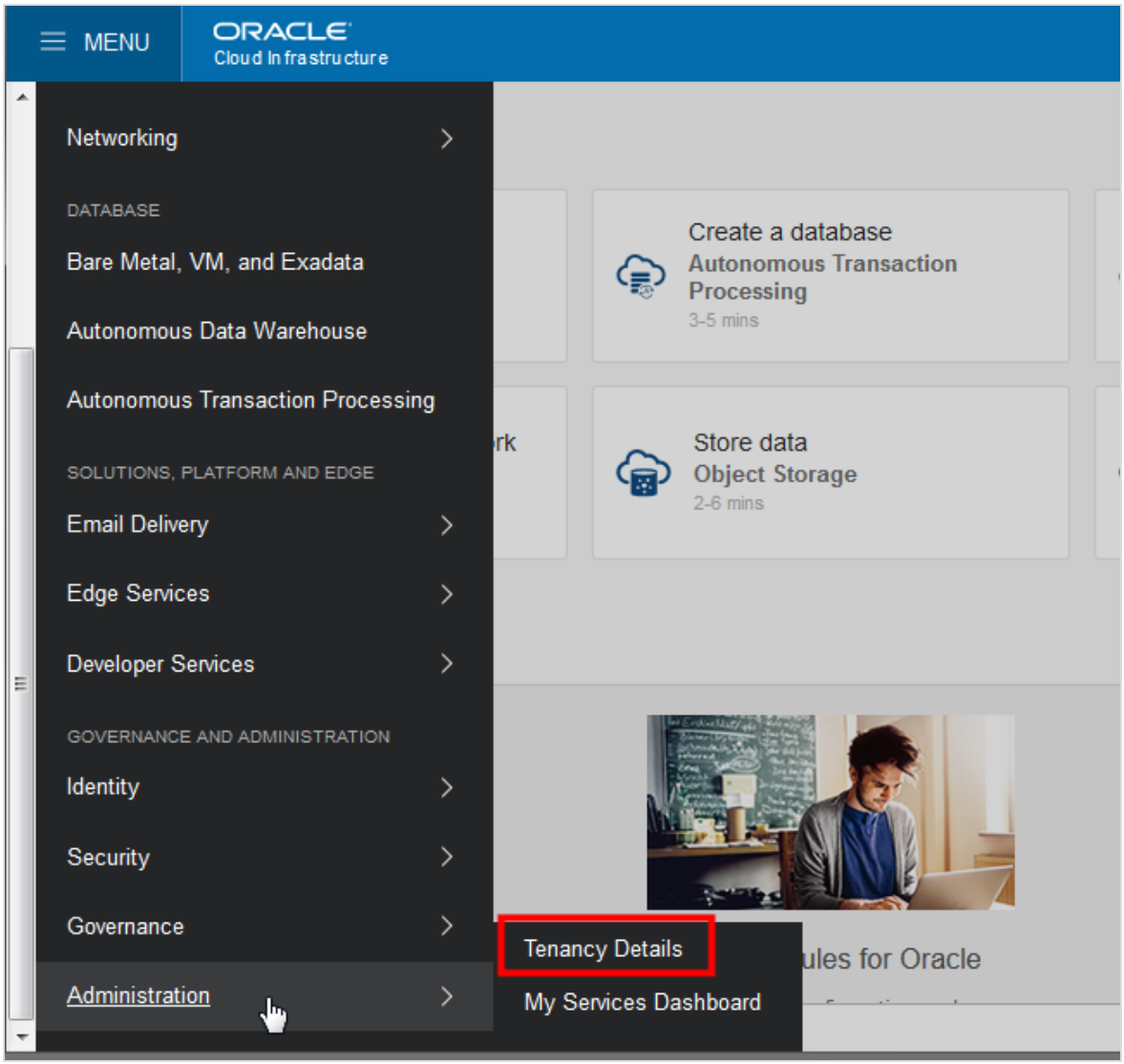
Where to Get the Tenancy's OCID and User's OCID

Both OCIDs are in the Console, which is located at https://console.us-ashburn-1.oraclecloud.com. If you don't have a login and password for the Console, contact an administrator. If you're not familiar with OCIDs, see Resource Identifiers.

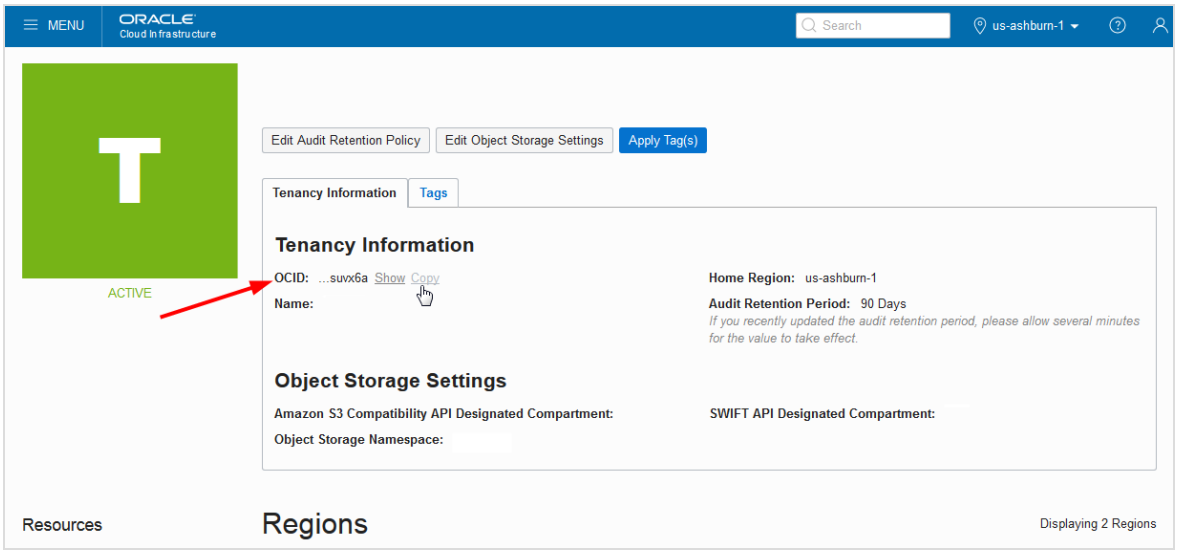
TENANCY'S OCID

Get the tenancy OCID from the Oracle Cloud Infrastructure Console on the **Tenancy Details** page:

1. Open the navigation menu, under Governance and Administration, go to **Administration** and click **Tenancy Details**.



2. The tenancy OCID is shown under **Tenancy Information**. Click **Copy** to copy it to your clipboard.



USER'S OCID

Get the user's OCID in the Console on the page showing the user's details. To get to that page:

- If you're signed in as the user: Open the **User** menu ( ) and click **User Settings**.
- If you're an administrator doing this for another user: Open the navigation menu. Under **Governance and Administration**, go to **Identity** and click **Users**. Select the user from the list.

How to Upload the Public Key

You can upload the PEM public key in the Console, located at https://console.us-ashburn-1.oraclecloud.com. If you don't have a login and password for the Console, contact an administrator.

1. Open the Console, and sign in.
2. View the details for the user who will be calling the API with the key pair:
    - If you're signed in as this user, click your username in the top-right corner of the Console, and then click **User Settings**.

- If you're an administrator doing this for another user, instead click **Identity**, click **Users**, and then select the user from the list.

3. Click **Add Public Key**.
4. Paste the contents of the PEM public key in the dialog box and click **Add**.

The key's fingerprint is displayed (for example, 12:34:56:78:90:ab:cd:ef:12:34:56:78:90:ab:cd:ef).

Notice that after you've uploaded your first public key, you can also use the UploadApiKey API operation to upload additional keys. You can have up to three API key pairs per user. In an API request, you specify the key's fingerprint to indicate which key you're using to sign the request.

SDK and CLI Configuration File

Oracle Cloud Infrastructure SDKs and CLI require basic configuration information, like user credentials and tenancy OCID. You can provide this information by:

- Using a configuration file
- Declaring a configuration at runtime

The SDKs fully support both options. Refer to the documentation for each SDK for information about the config object and any exceptions when using a configuration file:

- SDK for Java Configuration
- Python SDK Configuration
- Ruby SDK Configuration
- Go SDK Configuration

The CLI requires a configuration file. `--region` is the only configuration value that can be passed as a parameter for a CLI operation.

File Name and Location

The default configuration file name and location is `~/.oci/config`.



Note

On Windows, you can use PowerShell to create the folder with the following command: `mkdir ~/.oci`. File Explorer does not support creating folder names that start with a period.

File Entries

The following table lists the basic entries that are required for the configuration file, as well as where to get the required information.

Entry	Description and Where to Get the Value	Required?
`user`	OCID of the user calling the API. To get the value, see Required Keys and OCIDs. Example: `ocid1.user.oc1..aaaaaaaa65vwl75tewwm32rgqvm6i34unq` (shortened for brevity)	Yes
`fingerprint`	Fingerprint for the key pair being used. To get the value, see Required Keys and OCIDs. Example: `20:3b:97:13:55:1c:5b:0d:d3:37:d8:50:4e:c5:3a:34`	Yes

Entry	Description and Where to Get the Value	Required?
`key_file`	Full path and filename of the private key. **Important:** The key pair must be in PEM format. For instructions on generating a key pair in PEM format, see Required Keys and OCIDs. If you encrypted the key with a passphrase, you must also include the `pass_phrase` entry in the config file. Example: `~/.oci/oci_api_key.pem`	Yes
`pass_phrase`	Passphrase used for the key, if it is encrypted. Example: `examplephrase`	If key is encrypted
`tenancy`	OCID of your tenancy. To get the value, see Required Keys and OCIDs. Example: `ocid1.tenancy.oc1..aaaaaaaaba3pv6wuzr4h25vqstifsfdsq` (shortened for brevity)	Yes
`region`	An Oracle Cloud Infrastructure region. See Regions and Availability Domains. Example: `us-ashburn-1`	Yes

CUSTOM VALUES

Some Oracle Cloud Infrastructure SDKs support defining custom values in the configuration file. Refer to the documentation for each SDK for more information.



Warning

Avoid saving confidential information in the configuration file.

PROFILES AND INHERITANCE

You can create multiple profiles with different values for these entries, then you can specify which profile to load.

Some Oracle Cloud Infrastructure SDKs require a DEFAULT profile and support profile inheritance. This means that any value that isn't explicitly defined for a given profile is inherited from the DEFAULT profile. Refer to the documentation for each SDK for more information.

Example Configuration

The following example shows key values in a configuration file and how to set profiles for a SDK that supports profile inheritance.

```
[DEFAULT]
user=ocid1.user.oc1..aaaaaaaat5nvwcna5j6aqzjcaty5eqbb6qt2jvpkanghtgdaqedqw3rynjq
fingerprint=20:3b:97:13:55:1c:5b:0d:d3:37:d8:50:4e:c5:3a:34
key_file=~/.oci/oci_api_key.pem
tenancy=ocid1.tenancy.oc1..aaaaaaaaba3pv6wkcr4jqae5f15p2b2m2yt2j6rx32uzr4h25vqstifsfdsq
region=us-ashburn-1

[ADMIN_USER]
user=ocid1.user.oc1..aaaaaaaa65vwl7zut55hiavppn4nbfwyccuecuch5tewwm32rgqvm6i34unq
fingerprint=72:00:22:7f:d3:8b:47:a4:58:05:b8:95:84:31:dd:0e
key_file=keys/admin_key.pem
pass_phrase=mysecretphrase
```

REST APIs

The Oracle Cloud Infrastructure APIs are typical REST APIs that use HTTPS requests and responses. This topic describes basic information about using the APIs.



Warning

Oracle recommends that you avoid using string values that include confidential information in the Oracle Cloud Infrastructure API.

API Reference and Endpoints

For links to the Oracle Cloud Infrastructure API reference and a list of the regional API endpoints, see API Reference and Endpoints.

API Version

The base path of the endpoint includes the desired API version (for example, 20160918). Here's an example for a POST request to create a new VCN in the Ashburn region:

```
POST https://iaas.us-ashburn-1.oraclecloud.com/20160918/vcns
```

Request Signing Required

All Oracle Cloud Infrastructure API requests must be signed for authentication purposes. For information about the required credentials and how to sign the requests, see Request Signatures.

HTTPS and TLS 1.2 Required

All Oracle Cloud Infrastructure API requests must support HTTPS and SSL protocol TLS 1.2.

Maximum Allowed Client Clock Skew

HTTP status code 401 (NotAuthenticated) is returned if the client's clock is skewed more than 5 minutes from the server's. To determine the server's clock time, use this curl command

with the API endpoint:

```
curl -s --head <endpoint> | grep Date
```

For example:

```
curl -s --head https://iaas.us-phoenix-1.oraclecloud.com | grep Date
```

Request and Response Format

The Oracle Cloud Infrastructure APIs use standard HTTP requests and responses. Each may contain Oracle-specific headers for pagination, entity tags (ETags), and so on as described elsewhere in this topic and in the API documentation.

Each response includes a unique Oracle-assigned request ID (for example, bb3f3275-f356-462a-93c4-bf40fb82bb02) in the `opc-request-id` response header. If you need to contact Oracle about a particular request, please provide this request ID.

Many of the API operations require JSON in the request body or return JSON in the response body. The specific contents of the JSON are described in the API documentation for the individual operation. Notice that the JSON is not wrapped or labeled according to the operation's name or the object's name or type.



Note

Make sure to set the `Content-Type` header to `application/json` in your POST and PUT requests that contain JSON in the body.

Example CreateVcn Request

```
POST https://iaas.us-phoenix-1.oraclecloud.com/20160918/vcns
host: iaas.us-phoenix-1.oraclecloud.com
opc-retry-token: 239787fs987
Content-Type: application/json
HTTP headers required for authentication
Other HTTP request headers per the HTTP spec
```

```
{
  "compartmentId":
"ocid1.compartment.oc1..aaaaaaaauwjnv47knr7uuuvqar5bshnspi6xoxsfebh3vy72fi4swgrkvuvq",
  "displayName": "Apex Virtual Cloud Network",
  "cidrBlock": "172.16.0.0/16"
}
```

Example CreateVcn Response

```
200 OK
opc-request-id: 6c4d01a6-f764-4325-a3f8-720c8b5cae7b

{
   "id": "ocid1.vcn.oc1.phx.aaaaaaaa4ex5pqjtkjhdb4h4gcnko7vx5uto5puj5noa5awznsqpwjt3pqyq",
   "compartmentId":
"ocid1.compartment.oc1..aaaaaaaauwjnv47knr7uuuvqar5bshnspi6xoxsfebh3vy72fi4swgrkvuvq",
   "displayName": "Apex Virtual Cloud Network",
   "cidrBlock": "172.16.0.0/16"
   "defaultRouteTableId":
"ocid1.routetable.oc1.phx.aaaaaaaaba3pv6wkcr4jqae5f44n2b2m2yt2j6rx32uzr4h25vqstifsfdsq",
   "defaultSecurityListId":
"ocid1.securitylist.oc1.phx.aaaaaaaac6h4ckr3ncbxmvwinfvzxjbr7owu5hfzbvtu33kfe7hgscs5fjaq"
   "defaultDhcpOptionsId":
"ocid1.dhcpoptions.oc1.phx.aaaaaaaawglzn7s5sogyfznl25a4vxgu76c2hrgvzcd3psn6vcx33lzmu2xa"
   "state": "PROVISIONING",
   "timeCreated": "2016-07-22T17:43:01.389+0000"
}
```

Error Format

If a request results in an error, the response contains a standard HTTP response code with 4xx for client errors and 5xx for server errors. The body also includes JSON with an error code and a description of the error. For example:

```
{
    "code": "InvalidParameter",
    "message": "Description may not be empty; description size must be between 1 and 400"
}
```

For a list of common errors across all services, see API Errors.

Request Throttling

Oracle Cloud Infrastructure applies throttling to many API requests to prevent accidental or abusive use of resources. If you make too many requests too quickly, you might see some succeed and others fail. Oracle recommends that you implement an exponential back-off, starting from a few seconds to a maximum of 60 seconds. When a request fails due to throttling, the system returns response code 429 and the following error code and description:

```
{
      "code": "TooManyRequests",
      "message": "User-rate limit exceeded."
}
```

Polling for Resource Status

Most Oracle Cloud Infrastructure resources, such as compute instances, have lifecycles. In many cases, you want your code to wait until a resource or work request reaches a specific state, or a timeout is exceeded, before taking further action.

You can poll a resource to determine its state. For example, when you call GetInstance, the response body contains an instance resource that includes the `lifecycleState` attribute. You might want your code to wait until the instance's `lifecycleState` is RUNNING before proceeding.

Different resources take different amounts of time to transition between states. Therefore, the optimal frequency and duration parameters for a polling strategy can vary among resources. The Oracle Cloud Infrastructure SDK waiters use the following default strategy:

- Use an exponential back-off, starting from a few seconds to a maximum of 30 seconds between poll attempts.
- Poll up to 20 minutes, and then stop.

Or more information on waiters, see:

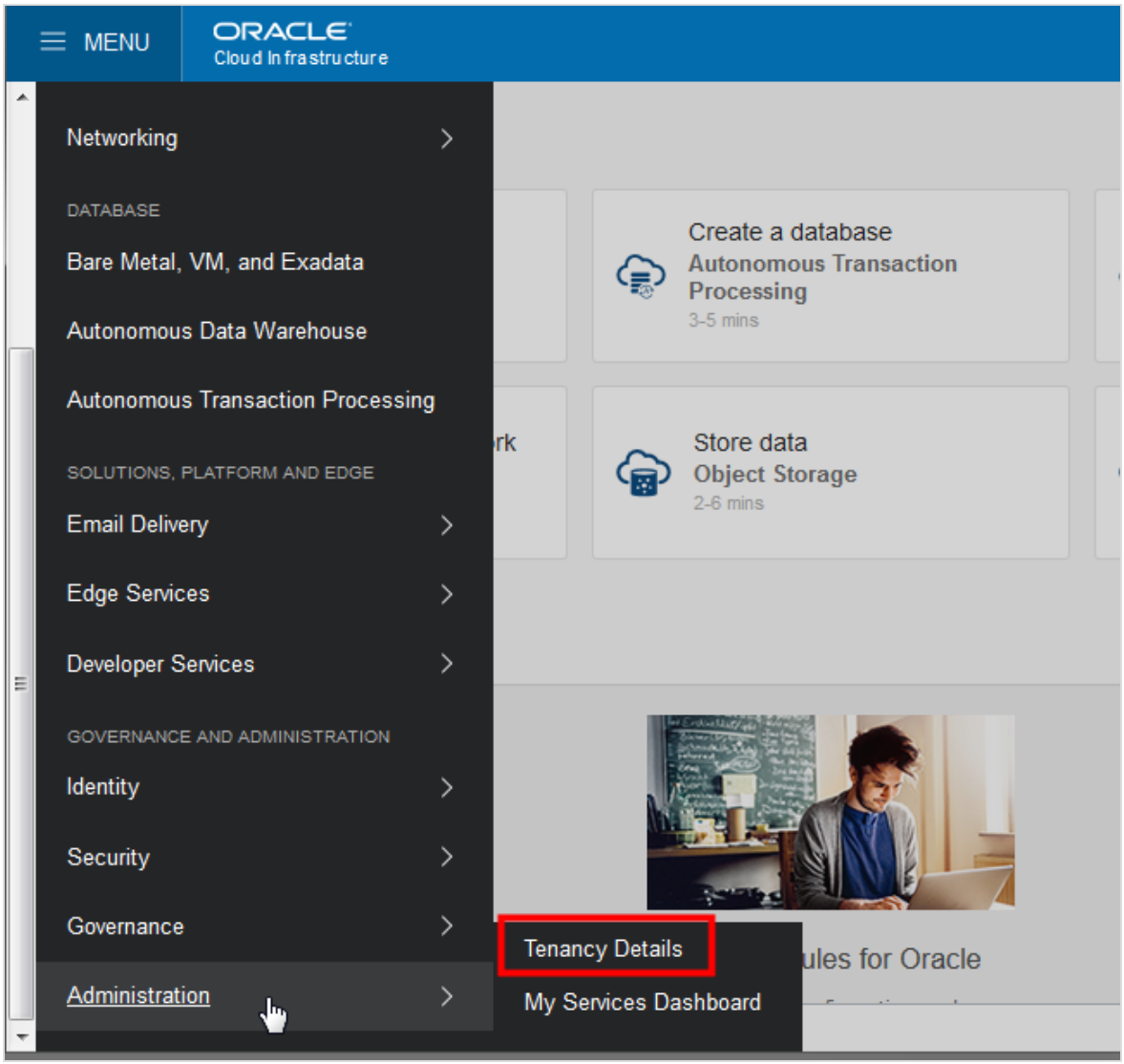
- SDK for Java waiters documentation
- Ruby SDK waiters documentation

Where to Find Your Tenancy's OCID

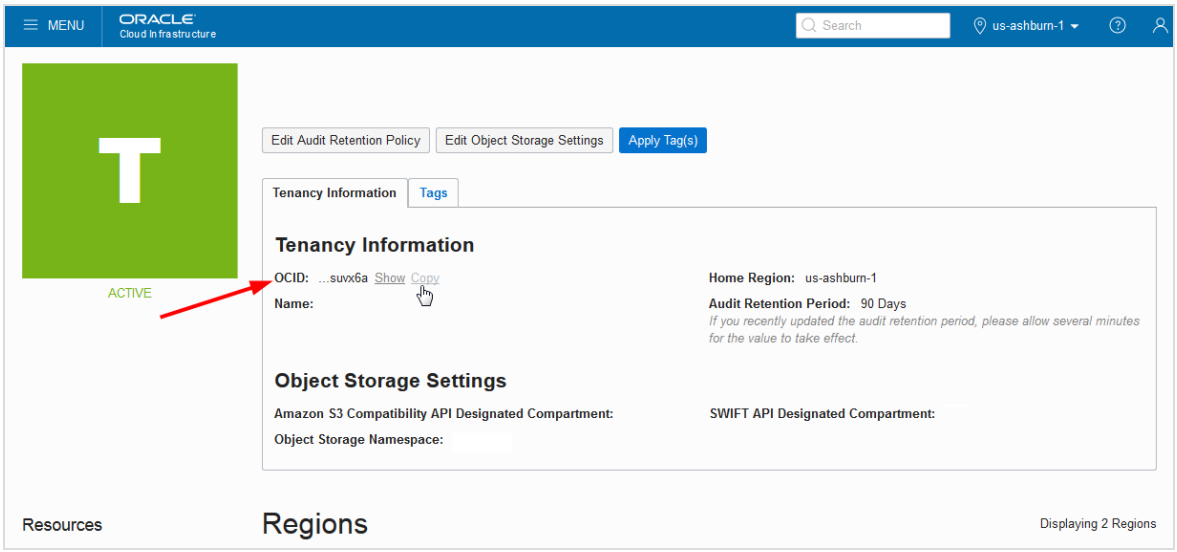
If you use the API, you'll need your tenancy's OCID in order to sign the requests (see Request Signatures). You'll also need it for some of the IAM API operations. An OCID is an Oracle Cloud ID (see Resource Identifiers).

Get the tenancy OCID from the Oracle Cloud Infrastructure Console on the **Tenancy Details** page:

1. Open the navigation menu, under Governance and Administration, go to **Administration** and click **Tenancy Details**.



2. The tenancy OCID is shown under **Tenancy Information**. Click **Copy** to copy it to your clipboard.



The tenancy OCID looks something like this (notice the word "tenancy" in it): `ocid1.tenancy.oc1..<unique_ID>`.

List Pagination

Most List operations paginate results. For example, results are paginated for the ListInstances operation in the Core Services API. When you call a paginated List operation, the response indicates additional pages of results by including the `opc-next-page` header.

> **Note**
>
> A page can be empty even when more results remain. Any time the `opc-next-page` header appears, there are more list items to get. For more information about resource list control, see Overview of Search.

To get the next page of results

Make a new GET request against the same URL, modified by setting the page query parameter to the value from the `opc-next-page` header. Repeat this process until you get a response without an `opc-next-page` header. The absence of this header indicates that you have reached the last page of the list.



Note

For an alternative to writing pagination code, see the functions in the pagination module provided with the Python SDK.

To get the previous page of results

(Available with some APIs.) Make a new GET request against the same URL, modified by setting the page query parameter to the value from the `opc-prev-page` header. Repeat this process until you get a response without an `opc-prev-page` header. The absence of this header indicates that you have reached the first page of the list.



Note

For an alternative to writing pagination code, see the functions in the pagination module provided with the Python SDK.

To change the maximum number of results per page

In the GET request, set the `limit` to the number of items you want returned in the response.



Note

The service will return no more than the number specified as `limit`, but might not return that exact number.

Retry Token

For some operations you can provide a unique retry token (`opc-retry-token`) so the request can be retried in case of a timeout or server error without the risk of executing that same action again. The token expires after 24 hours, but can be invalidated before then due to conflicting operations (for example, if a resource has been deleted and purged from the system, then a retry of the original creation request may be rejected).

ETags for Optimistic Concurrency Control

The API supports etags for the purposes of optimistic concurrency control. The GET and POST calls return an `etag` response header with a value you should store. When you later want to update or delete the resource, set the `if-match` header to the ETag you received for the resource. The resource will then be updated or deleted *only* if the ETag you provide matches the current value of that resource's ETag.

Null vs. Empty Strings for Optional Parameters

If you send an empty string ("") as the value of an optional parameter, the API validates the value as normal (for example, checks against minimum and maximum allowed length, and so on). Often the minimum allowed length is 1, so an error would be returned. If you don't set the value (it's null), the API performs no validation, and some other action may occur. For example: if you don't set a value for the `displayName` when creating a new VCN object, the service will auto-generate a value.

API Reference and Endpoints

Oracle Cloud Infrastructure has these APIs and corresponding regional endpoints:

Announcements API

API reference

- https://announcements.ap-seoul-1.oraclecloud.com
- https://announcements.ap-tokyo-1.oraclecloud.com
- https://announcements.ca-toronto-1.oraclecloud.com
- https://announcements.eu-frankfurt-1.oraclecloud.com
- https://announcements.uk-london-1.oraclecloud.com
- https://announcements.us-ashburn-1.oraclecloud.com
- https://announcements.us-phoenix-1.oraclecloud.com

Audit API

API reference

- https://audit.ap-seoul-1.oraclecloud.com
- https://audit.ap-tokyo-1.oraclecloud.com
- https://audit.ca-toronto-1.oraclecloud.com
- https://audit.eu-frankfurt-1.oraclecloud.com
- https://audit.uk-london-1.oraclecloud.com
- https://audit.us-ashburn-1.oraclecloud.com
- https://audit.us-phoenix-1.oraclecloud.com

Budgets API

API reference

- https://usage.ap-seoul-1.oci.oraclecloud.com
- https://usage.ap-tokyo-1.oci.oraclecloud.com
- https://usage.ca-toronto-1.oci.oraclecloud.com
- https://usage.eu-frankfurt-1.oci.oraclecloud.com
- https://usage.uk-london-1.oci.oraclecloud.com
- https://usage.us-ashburn-1.oci.oraclecloud.com
- https://usage.us-phoenix-1.oci.oraclecloud.com

Container Engine for Kubernetes API

API reference

- https://containerengine.ap-seoul-1.oraclecloud.com
- https://containerengine.ap-tokyo-1.oraclecloud.com
- https://containerengine.ca-toronto-1.oraclecloud.com
- https://containerengine.eu-frankfurt-1.oraclecloud.com
- https://containerengine.uk-london-1.oraclecloud.com
- https://containerengine.us-ashburn-1.oraclecloud.com
- https://containerengine.us-phoenix-1.oraclecloud.com

Core Services (covering Networking, Compute, and Block Volume)

The Networking, Compute, and Block Volume services are accessible with the following API:

Core Services API

API reference

- https://iaas.ap-seoul-1.oraclecloud.com
- https://iaas.ap-tokyo-1.oraclecloud.com
- https://iaas.ca-toronto-1.oraclecloud.com
- https://iaas.eu-frankfurt-1.oraclecloud.com
- https://iaas.uk-london-1.oraclecloud.com
- https://iaas.us-ashburn-1.oraclecloud.com
- https://iaas.us-phoenix-1.oraclecloud.com

You can also manage Compute instances with the following API:

Autoscaling API

API reference

- https://autoscaling.ca-toronto-1.oci.oraclecloud.com
- https://autoscaling.eu-frankfurt-1.oci.oraclecloud.com
- https://autoscaling.uk-london-1.oci.oraclecloud.com
- https://autoscaling.us-ashburn-1.oci.oraclecloud.com
- https://autoscaling.us-phoenix-1.oci.oraclecloud.com

Database API

API reference

- https://database.ap-seoul-1.oraclecloud.com
- https://database.ap-tokyo-1.oraclecloud.com
- https://database.ca-toronto-1.oraclecloud.com
- https://database.eu-frankfurt-1.oraclecloud.com
- https://database.uk-london-1.oraclecloud.com

- https://database.us-ashburn-1.oraclecloud.com
- https://database.us-phoenix-1.oraclecloud.com

DNS API

API reference

- https://dns.ap-seoul-1.oraclecloud.com
- https://dns.ap-tokyo-1.oraclecloud.com
- https://dns.ca-toronto-1.oraclecloud.com
- https://dns.eu-frankfurt-1.oraclecloud.com
- https://dns.uk-london-1.oraclecloud.com
- https://dns.us-ashburn-1.oraclecloud.com
- https://dns.us-phoenix-1.oraclecloud.com

Email Delivery API

API reference

- https://email.us-ashburn-1.oraclecloud.com
- https://email.us-phoenix-1.oraclecloud.com

File Storage API

API reference

- https://filestorage.ap-seoul-1.oraclecloud.com
- https://filestorage.ap-tokyo-1.oraclecloud.com
- https://filestorage.ca-toronto-1.oraclecloud.com
- https://filestorage.eu-frankfurt-1.oraclecloud.com

- https://filestorage.uk-london-1.oraclecloud.com
- https://filestorage.us-ashburn-1.oraclecloud.com
- https://filestorage.us-phoenix-1.oraclecloud.com

Health Checks API

API reference

- https://healthchecks.ca-toronto-1.oraclecloud.com
- https://healthchecks.eu-frankfurt-1.oraclecloud.com
- https://healthchecks.uk-london-1.oraclecloud.com
- https://healthchecks.us-ashburn-1.oraclecloud.com
- https://healthchecks.us-phoenix-1.oraclecloud.com

IAM API

API reference

- https://identity.ap-seoul-1.oraclecloud.com
- https://identity.ap-tokyo-1.oraclecloud.com
- https://identity.ca-toronto-1.oraclecloud.com
- https://identity.eu-frankfurt-1.oraclecloud.com
- https://identity.uk-london-1.oraclecloud.com
- https://identity.us-ashburn-1.oraclecloud.com
- https://identity.us-phoenix-1.oraclecloud.com



Note

Use the Endpoint of Your Home Region for All IAM API Calls

When you sign up for Oracle Cloud Infrastructure, Oracle creates a tenancy for you in one region. This is your home region. Your home region is where your IAM resources are defined. When you subscribe to a new region, your IAM resources are replicated in the new region, however, the master definitions reside in your home region and can only be changed there. Make all IAM API calls against your home region endpoint. The changes automatically replicate to all regions. If you try to make an IAM API call against a region that is not your home region, you will receive an error.

Key Management API

API reference

- https://kms.ap-seoul-1.oraclecloud.com
- https://kms.ap-tokyo-1.oraclecloud.com
- https://kms.ca-toronto-1.oraclecloud.com
- https://kms.eu-frankfurt-1.oraclecloud.com
- https://kms.uk-london-1.oraclecloud.com
- https://kms.us-ashburn-1.oraclecloud.com
- https://kms.us-phoenix-1.oraclecloud.com

In addition to these endpoints, each vault has a unique endpoint for create, update, and list operations for keys. This endpoint is referred to as the control plane URL or management

endpoint. Each vault also has a unique endpoint for cryptographic operations. This endpoint is known as the data plane URL or the cryptographic endpoint.

Load Balancing API

API reference

- https://iaas.ap-seoul-1.oraclecloud.com
- https://iaas.ap-tokyo-1.oraclecloud.com
- https://iaas.ca-toronto-1.oraclecloud.com
- https://iaas.eu-frankfurt-1.oraclecloud.com
- https://iaas.uk-london-1.oraclecloud.com
- https://iaas.us-ashburn-1.oraclecloud.com
- https://iaas.us-phoenix-1.oraclecloud.com

Monitoring

API reference

PostMetricData operation:

- https://telemetry-ingestion.ap-tokyo-1.oraclecloud.com
- https://telemetry-ingestion.ca-toronto-1.oraclecloud.com
- https://telemetry-ingestion.eu-frankfurt-1.oraclecloud.com
- https://telemetry-ingestion.uk-london-1.oraclecloud.com
- https://telemetry-ingestion.us-ashburn-1.oraclecloud.com
- https://telemetry-ingestion.us-phoenix-1.oraclecloud.com

All other operations:

- https://telemetry.ap-tokyo-1.oraclecloud.com
- https://telemetry.ca-toronto-1.oraclecloud.com
- https://telemetry.eu-frankfurt-1.oraclecloud.com
- https://telemetry.uk-london-1.oraclecloud.com
- https://telemetry.us-ashburn-1.oraclecloud.com
- https://telemetry.us-phoenix-1.oraclecloud.com

Notifications

API reference

- https://cp.notification.ap-tokyo-1.oraclecloud.com
- https://cp.notification.ca-toronto-1.oraclecloud.com
- https://cp.notification.eu-frankfurt-1.oraclecloud.com
- https://cp.notification.uk-london-1.oraclecloud.com
- https://cp.notification.us-ashburn-1.oraclecloud.com
- https://cp.notification.us-phoenix-1.oraclecloud.com

Object Storage and Archive Storage

Both Object Storage and Archive Storage are accessible with the following APIs:

Object Storage API

API reference

- https://objectstorage.ap-seoul-1.oraclecloud.com
- https://objectstorage.ap-tokyo-1.oraclecloud.com
- https://objectstorage.ca-toronto-1.oraclecloud.com

- https://objectstorage.eu-frankfurt-1.oraclecloud.com
- https://objectstorage.uk-london-1.oraclecloud.com
- https://objectstorage.us-ashburn-1.oraclecloud.com
- https://objectstorage.us-phoenix-1.oraclecloud.com

Amazon S3 Compatibility API

API reference

- https://*<object_storage_namespace>*.compat.objectstorage.ap-seoul-1.oraclecloud.com
- https://*<object_storage_namespace>*.compat.objectstorage.ap-tokyo-1.oraclecloud.com
- https://*<object_storage_namespace>*.compat.objectstorage.ca-toronto-1.oraclecloud.com
- https://*<object_storage_namespace>*.compat.objectstorage.eu-frankfurt-1.oraclecloud.com
- https://*<object_storage_namespace>*.compat.objectstorage.uk-london-1.oraclecloud.com
- https://*<object_storage_namespace>*.compat.objectstorage.us-ashburn-1.oraclecloud.com
- https://*<object_storage_namespace>*.compat.objectstorage.us-phoenix-1.oraclecloud.com



Tip

See Understanding Object Storage Namespaces for information regarding how to find your Object Storage namespace.

Swift API (for use with Oracle RMAN)

- https://swiftobjectstorage.ap-seoul-1.oraclecloud.com
- https://swiftobjectstorage.ap-tokyo-1.oraclecloud.com
- https://swiftobjectstorage.ca-toronto-1.oraclecloud.com
- https://swiftobjectstorage.eu-frankfurt-1.oraclecloud.com
- https://swiftobjectstorage.uk-london-1.oraclecloud.com
- https://swiftobjectstorage.us-ashburn-1.oraclecloud.com
- https://swiftobjectstorage.us-phoenix-1.oraclecloud.com

Oracle Cloud My Services API

API reference

- https://itra.oraclecloud.com/

Resource Manager API

API reference

- https://resourcemanager.ca-toronto-1.oraclecloud.com/
- https://resourcemanager.eu-frankfurt-1.oraclecloud.com/
- https://resourcemanager.uk-london-1.oraclecloud.com/

- https://resourcemanager.us-ashburn-1.oraclecloud.com/
- https://resourcemanager.us-phoenix-1.oraclecloud.com/

Search API

API reference

- https://query.ap-seoul-1.oraclecloud.com/
- https://query.ap-tokyo-1.oraclecloud.com/
- https://query.ca-toronto-1.oraclecloud.com/
- https://query.eu-frankfurt-1.oraclecloud.com/
- https://query.uk-london-1.oraclecloud.com/
- https://query.us-ashburn-1.oraclecloud.com/
- https://query.us-phoenix-1.oraclecloud.com/

Streaming API

API reference

- https://streams.ap-tokyo-1.streaming.oci.oraclecloud.com
- https://streams.ca-toronto-1.streaming.oci.oraclecloud.com
- https://streams.eu-frankfurt-1.streaming.oci.oraclecloud.com
- https://streams.uk-london-1.streaming.oci.oraclecloud.com
- https://streams.us-ashburn-1.streaming.oci.oraclecloud.com
- https://streams.us-phoenix-1.streaming.oci.oraclecloud.com

Web Application Acceleration and Security API

API reference

- https://waas.ca-toronto-1.oraclecloud.com
- https://waas.eu-frankfurt-1.oraclecloud.com
- https://waas.uk-london-1.oraclecloud.com
- https://waas.us-ashburn-1.oraclecloud.com
- https://waas.us-phoenix-1.oraclecloud.com

API Errors

Common Errors Returned by All Services

The following table lists the common errors returned by all the services for Oracle Cloud Infrastructure.

HTTP Status Code	Error Code	Description	Retry
400	`CannotParseRequest`	The request is incorrectly formatted.	No.
400	`InvalidParameter`	A parameter is invalid or incorrectly formatted.	No.
400	`LimitExceeded`	Fulfilling this request exceeds the Oracle-defined limit for this tenancy for this resource type.	No.
400	`MissingParameter`	A required parameter is missing.	No.
400	`QuotaExceeded`	Fulfilling this request exceeds the administrator-defined quota for this compartment for this resource.	No.

HTTP Status Code	Error Code	Description	Retry
400	`RelatedResourceNotAuthorizedOrNotFound`	A resource specified in the body of the request was not found, or you do not have authorization to access that resource.	Yes, with backoff.
401	`NotAuthenticated`	The required authentication information was not provided or was incorrect. There are other reasons why this error code is generated. For more information, see HTML Status Code 401.	Yes, with backoff.
402	`SignUpRequired`	This operation requires opt-in before it may be called.	No.
404	`NotAuthorizedOrNotFound`	A resource specified via the URI (path or query parameters) of the request was not found, or you do not have authorization to access that resource. For more information, see HTML Status Code 404.	Yes, with backoff.
404	`NotFound`	There is no operation supported at the URI path and HTTP method you specified in the request.	No.
409	`IncorrectState`	The requested state for the resource conflicts with its current state.	Yes, with backoff.

HTTP Status Code	Error Code	Description	Retry
409	`InvalidatedRetryToken`	The provided retry token was used in an earlier request that resulted in a system update, but a subsequent operation invalidated the token. This can happen, for example, in cases where an entity created with the same token has since been deleted. If the system state change that is associated with this request should be performed again, retry it using a different token.	No.
409	`NotAuthorizedOr ResourceAlreadyExists`	You do not have authorization to perform this request, or the resource you are attempting to create already exists. This error code is returned only from **create** operations, where it is returned instead of the more general **NotAuthorizedOrNotFound** error code."	Yes, with backoff.
412	`NoEtagMatch`	The ETag specified in the request does not match the ETag for the resource.	No.
429	`TooManyRequests`	You have issued too many requests to the Oracle Cloud Infrastructure APIs in too short of an amount of time.	Yes, with backoff.
500	`InternalServerError`	An internal server error occurred.	Yes, with backoff.

Error Details and Troubleshooting

HTTP STATUS CODE: 401

- **Missing or incorrect authentication information.** Verify that all the required information (tenant OCID, user OCID, fingerprint, and private key) is provided and accurate. Verify that the public key corresponding to the fingerprint has been uploaded for the user. For more information, see Required Keys and OCIDs.
- **Clock skew.** This status code is returned if the client's clock is skewed more than five (5) minutes from the server's clock. For more information, see Maximum Allowed Client Clock Skew.
- **API request signature error.** This status code is returned if a required header is missing from a signing string. For more information, see Request Signatures.

ERROR CODES: NOTAUTHORIZEDORNOTFOUND, RELATEDRESOURCENOTAUTHORIZEDORNOTFOUND , NOTAUTHORIZEDORRESOURCEALREADYEXISTS

- **Authorization error.** Verify that the user is in a group that has the permissions to work with resources in a compartment.
- **Compartment or resource not found.** Verify that the compartment or resource exist and is referenced correctly. For example, this status code is returned for either of the following errors:
  - `CompartmentNotFound` if a compartment doesn't exist
  - `VolumeNotFound` if a volume doesn't exist

Asynchronous Work Requests

This topic describes asynchronous work requests for long-running operations against Oracle Cloud Infrastructure services. It also provides guidance on obtaining request status, and for inspecting the request response to enable filtering for affected resources.

Overview

API calls to Oracle Cloud Infrastructure services can launch long-running operations that do not complete the client's request before a response is returned. In these cases, the service spawns an *asynchronous work request* that allows for programmatic visibility into the progress of long-running, asynchronous operations. The response to the REST API call contains a work request ID in the `opc-work-request-id` header, which allows you to monitor its progress and status. The work request itself remains in a queue until the operation has completed.

You can monitor the status of the work request at any time by calling `GetWorkRequest` and passing in the work request ID.



Note

Presently, Oracle Cloud Infrastructure supports work requests on three services: Load Balancing, Object Storage, and Container Engine for Kubernetes. Each of these services exposes its own `GetWorkRequest` API for fetching the details of a `WorkRequest`.

Two features of the request response are of particular interest: the status of the work request, and a list of the resources that are affected by the work request. The status is important because asynchronous work requests must know when an operation has completed, is still running, or whether it has failed altogether.

To retrieve information about work request failures or errors, each service provides APIs for fetching information about errors, and logs. For links to API reference documentation for each of the service, see the section For More Information.

Also important in cases where a work request operation affects several resources is having a list of the resources that a work request affects, along with each one's `entityType` and `actionType` attributes.

Work Request Status

Asynchronous work requests allow you to monitor their progress by providing a status attribute on the `WorkRequest` object. Each of the supported services provides its own API for obtaining status, as listed in the following sections.



Note

There is a ContainEngineWaiters class that allows you to create a callback using the `forWorkRequest` method. Use this API to forward a notification when an operation's status changes, for example, from `IN_PROGRESS` to `COMPLETED`.

The following table lists status attributes that are supported by the `WorkRequest` object on the respective services.

Service	Status Attributes
Object Storage	• ACCEPTED • IN_PROGRESS • FAILED • COMPLETED • CANCELING • CANCELED
Container Engine for Kubernetes	• ACCEPTED • IN_PROGRESS • FAILED • SUCCEEDED • CANCELING • CANCELED
Load Balancing	• CREATING • FAILED • ACTIVE • DELETING • DELETED

Filtering the Request Response

You sometimes need to know which resources are affected by a given asynchronous work request. In cases where the request response includes just one or two affected resources, the body of the request response is probably sufficient. However, in cases where a request response affects a great many resources, you must filter the response to identify the resources that you're interested in.

Filtering of resources listed in a work request response relies on two attributes of the `WorkRequestResource` type: `entityType` and `actionType`.

- **entityType**: Represents the resource type which the work request affects. This is an optional attribute, but each resource can have only one `entityType`.
- **actionType**: Represents how the specified resource is affected by the operation associated with the work request. Each service specifies a fixed list of allowable `actionType` values (shown in the sections following).

To obtain resource information on a work request, call `GetWorkRequest` and pass in the work request ID. The call returns a response in JSON format. Following is an example.

```
{
    operationType: "COPY_OBJECT",
    status: "IN_PROGRESS",
    id: "f54527d6-029b-4221-9046-a811b7686202",
    resources: [
        {
            entityType: "object",
            actionType: "READ",
            entityUri: "/n/mynamespace/b/backups/o/myobject"
        },
        {
            entityType: "object",
            actionType: "WRITTEN",
            entityUri: "/n/mynamespace/b/backups/o/copyofmyobject"
        },
    ],
    timeAccepted: 2017-10-13T17:23:46.000Z,
    timeStarted: 2017-10-13T17:23:52.198Z,
    percentComplete: 10.0
}
```



Note

The example is from an Object Storage service work request. Different services provide slightly different responses. See the reference documentation for each service's work request API for details. Links to each are provided in the "For More Information" section.

The following table lists the entity types and actions types that are supported by Oracle Cloud Infrastructure services.

Service Name	Operation	entityType	actionType
Object Storage	CopyObject	`object`	`READ` `WRITTEN`
Container Engine for Kubernetes	CreateCluster DeleteCluster UpdateCluster CreateNodepool DeleteNodePool UpdateNodePool	`cluster` `nodepool`	`ACCEPTED` `IN_PROGRESS` `FAILED` `SUCCEEDED` `CANCELING` `CANCELED`
Load Balancing	CreateLoadBalancer UpdateLoadBalancer DeleteLoadBalancer	`LoadBalancer`	`ACCEPTED` `IN_PROGRESS` `FAILED` `SUCCEEDED`

Request/Response Sample

Following is a sequence of REST API calls to create a cluster, which is a common long-running operation. The caller retrieves the work request id from the response to the initial `POST` call and then periodically polls the `WorkRequest` to determine the status of the operation. The request/response sequence that follows depicts this workflow:

1. The user issues a `CreatCluster` API call.
2. The service responds with status code 202, indicating that the request has been accepted and returns a work request ID in the `Opc-Work-Request-Id` header.
3. Next, the user issues a `GET` call on the work request ID to obtain the status of the work request.
4. The service responds with status code 200, indicating in the response body that the `CLUSTER_CREATE` operation has the status `ACCEPTED`.
5. With continued polling, we see another `GET` call for the work request.
6. The service responds with status code 200. The response body reports that the operation `SUCCEEDED`.

Step 1. Initial API call to initiate a `CLUSTER_CREATE` operation.

```
POST https://containerengine.eu-frankfurt-1.oraclecloud.com/20180222/clusters
Accept: application/json
authorization: <Redacted>
content-length: 480
Content-Type: application/json
date: Mon, 02 Jul 2018 18:20:03 GMT
host: containerengine.eu-frankfurt-1.oraclecloud.com
opc-client-info: Oracle-JavaSDK/1.2.42-preview1-SNAPSHOT
opc-request-id: D7A390ED909C47038C438BA3629FB612
User-Agent: Oracle-JavaSDK/1.2.42-preview1-SNAPSHOT (Mac OS X/10.13.5; Java/1.8.0_172; Java HotSpot(TM)
64-Bit Server VM/25.172-b11)
x-content-sha256: S8U8OKQHyTLNViAzgexkjxvF4ctncJJHTjuRfXn0ya4={
   "name":"JavaSDK.CRUD",

"compartmentId":"ocid1.compartment.oc1..aaaaaaaa4k5c6anmwmbo7iqki5rbbbtbrtxyyeml3g5pikhavg6yuv4ntqvq",
   "vcnId":"ocid1.vcn.oc1.eu-frankfurt-1.aaaaaaaa3yq5p4a27hdx6iwoxzasedooexlypn7a2eyhraspgbmkhst4i2sa",
   "kubernetesVersion":"v1.10.3",
   "options":{"serviceLbSubnetIds":["ocid1.subnet.oc1.eu-frankfurt-
```

```
1.aaaaaaaa5653tiavvsdgkkfkop534zkw76tpzsse6iqb5tvpw3huowlgdpyq",
   "ocid1.subnet.oc1.eu-frankfurt-1.aaaaaaaa3mnjfbo35wc2qqyfustjcxcybagastnb5j2pmbucs2wuuy2hjfzq"]}}
```

Step 2. The response to the initial API call, which contains the work request ID in the `Opc-Work-Request-Id header`.

```
202
Access-Control-Allow-Methods: DELETE, GET, HEAD, OPTIONS, PATCH, POST, PUT
Access-Control-Allow-Origin: *
Access-Control-Expose-Headers: opc-work-request-id
Content-Length: 0
Date: Mon, 02 Jul 2018 18:20:04 GMT
Opc-Request-Id:
D7A390ED909C47038C438BA3629FB612/33EEDCAAB2E84508B34AA75CD0FD86F4/8261D1CC89814E9BB934440A1F43DA09
Opc-Work-Request-Id: ocid1.clustersworkrequest.oc1.eu-frankfurt-
1.aaaaaaaaaftdqojrmq3tcobtgrswkytchezdinjxgmzdkmrwhwtginrxgnsw
Uri: /20180222/clusters
Vary: Accept-Encoding
X-Rate-Limit-Duration: 1
X-Rate-Limit-Limit: 16.70
X-Rate-Limit-Request-Forwarded-For: 10.237.10.0, 10.237.9.51
X-Rate-Limit-Request-Remote-Addr: 10.237.9.51:53077
```

Step 3. Since this is a long-running operation, the user periodically polls the work request using a `GET` call to determine its status.

```
GET https://containerengine.eu-frankfurt-
1.oraclecloud.com/20180222/workRequests/ocid1.clustersworkrequest.oc1.eu-frankfurt-
1.aaaaaaaaaftdqojrmq3tcobtgrswkytchezdinjxgmzdkmrwhwtginrxgnsw
Accept: application/json
authorization: <Redacted>
date: Mon, 02 Jul 2018 18:20:04 GMT
host: containerengine.eu-frankfurt-1.oraclecloud.com
opc-client-info: Oracle-JavaSDK/1.2.42-preview1-SNAPSHOT
opc-request-id: E8F20DAC443346B3B0EA599F367EE294
User-Agent: Oracle-JavaSDK/1.2.42-preview1-SNAPSHOT (Mac OS X/10.13.5; Java/1.8.0_172; Java HotSpot(TM)
64-Bit Server VM/25.172-b11)
```

Step 4. The `GET` call returns the following response, which indicates in the response body that the `CLUSTER_CREATE` operation has a status of `ACCEPTED`.

```
200
Access-Control-Allow-Methods: DELETE, GET, HEAD, OPTIONS, PATCH, POST, PUT
Access-Control-Allow-Origin: *
```

```
Access-Control-Expose-Headers: opc-work-request-id
Content-Length: 717
Content-Type: application/json
Date: Mon, 02 Jul 2018 18:20:05 GMT
Etag: 56a41efaf33d81a54933495ee910c24d7bce7a83adf18810f95e07bdd2055805
Opc-Request-Id:
E8F20DAC443346B3B0EA599F367EE294/8B19C9FC3B4442CEA14685D1973D0856/0BA60B0711764DE4A4373071632708C7
Retry-After: 30
Uri: /20180222/workRequests/_id_
Vary: Accept-Encoding
X-Rate-Limit-Duration: 1
X-Rate-Limit-Limit: 16.70
X-Rate-Limit-Request-Forwarded-For: 10.237.10.0, 10.237.9.51
X-Rate-Limit-Request-Remote-Addr: 10.237.9.51:43533
{
  "id": "ocid1.clustersworkrequest.oc1.eu-frankfurt-
1.aaaaaaaaaftdqojrmq3tcobtgrswkytchezdinjxgmzdkmrwhwtginrxgnsw",
  "operationType": "CLUSTER_CREATE",
  "status": "ACCEPTED",
  "compartmentId":
"ocid1.compartment.oc1..aaaaaaaa4k5c6anmwmbo7iqki5rbbbtbrtxyyeml3g5pikhavg6yuv4ntqvq",
  "resources": [
    {
      "actionType": "IN_PROGRESS",
      "entityType": "cluster",
      "identifier": "ocid1.cluster.oc1.eu-frankfurt-
1.aaaaaaaaaftdgobuha4geojsgu2tgmbxmy3dknzzg42dcnrvmcrgimzygbrg",
      "entityUri": "/clusters/ocid1.cluster.oc1.eu-frankfurt-
1.aaaaaaaaaftdgobuha4geojsgu2tgmbxmy3dknzzg42dcnrvmcrgimzygbrg"
    }
  ],
  "timeAccepted": "2018-07-02T18:20:05Z",
  "timeStarted": null,
  "timeFinished": null
}
```

Step 5. The operation continues, and the user continues to poll the work request using the `GET` method.

```
GET https://containerengine.eu-frankfurt-
1.oraclecloud.com/20180222/workRequests/ocid1.clustersworkrequest.oc1.eu-frankfurt-
1.aaaaaaaaaftdqojrmq3tcobtgrswkytchezdinjxgmzdkmrwhwtginrxgnsw
Accept: application/json
```

```
authorization: <Redacted>
date: Mon, 02 Jul 2018 18:24:13 GMT
host: containerengine.eu-frankfurt-1.oraclecloud.com
opc-client-info: Oracle-JavaSDK/1.2.42-preview1-SNAPSHOT
opc-request-id: 64595B97E39A471A886DA29966BB6B1D
User-Agent: Oracle-JavaSDK/1.2.42-preview1-SNAPSHOT (Mac OS X/10.13.5; Java/1.8.0_172; Java HotSpot(TM)
64-Bit Server VM/25.172-b11)
```

Step 6. The last `GET` call produced the following response, which indicates that the operation has completed. Note the `entityType` is "cluster" and the `actionType` is "CREATED".

```
200
Access-Control-Allow-Methods: DELETE, GET, HEAD, OPTIONS, PATCH, POST, PUT
Access-Control-Allow-Origin: *
Access-Control-Expose-Headers: opc-work-request-id
Content-Length: 750
Content-Type: application/json
Date: Mon, 02 Jul 2018 18:24:14 GMT
Etag: 023d2a8ccb6d893fa8c875f64652353f21d22607825f49eeeb15b5394ae24918
Opc-Request-Id:
64595B97E39A471A886DA29966BB6B1D/3A81140991C94794AF365016E31DBE82/6245FBD8C25842B6BDF15187EA6ADB21
Uri: /20180222/workRequests/_id_
Vary: Accept-Encoding
X-Rate-Limit-Duration: 1
X-Rate-Limit-Limit: 16.70
X-Rate-Limit-Request-Forwarded-For: 10.237.3.0, 10.237.40.183
X-Rate-Limit-Request-Remote-Addr: 10.237.40.183:55856

{
  "id": "ocid1.clustersworkrequest.oc1.eu-frankfurt-
1.aaaaaaaaaftdqojrmq3tcobtgrswkytchezdinjxgmzdkmrwhwtginrxgnsw",
  "operationType": "CLUSTER_CREATE",
  "status": "SUCCEEDED",
  "compartmentId":
"ocid1.compartment.oc1..aaaaaaaa4k5c6anmwmbo7iqki5rbbbtbrtxyyeml3g5pikhavg6yuv4ntqvq",
  "resources": [
    {
      "actionType": "CREATED",
      "entityType": "cluster",
      "identifier": "ocid1.cluster.oc1.eu-frankfurt-
1.aaaaaaaaaftdgobuha4geojsgu2tgmbxmy3dknzzg42dcnrvmcrgimzygbrg",
      "entityUri": "/clusters/ocid1.cluster.oc1.eu-frankfurt-
1.aaaaaaaaaftdgobuha4geojsgu2tgmbxmy3dknzzg42dcnrvmcrgimzygbrg"
```

```
    }
  ],
  "timeAccepted": "2018-07-02T18:20:05Z",
  "timeStarted": "2018-07-02T18:20:10Z",
  "timeFinished": "2018-07-02T18:24:01Z"
}
```

For More Information

- Viewing the State of a Work Request
- WorkRequest API (Load Balancer)
- WorkRequest API (Object Storage)
- WorkRequest API (Container Engine)

Request Signatures

This topic describes how to sign Oracle Cloud Infrastructure API requests.

Signing samples are included for the following:

- Bash
- PowerShell
- C#
- Java
- NodeJS
- Perl
- PHP
- Python
- Ruby
- Go

Signature Version 1

The signature described here is *version 1* of the Oracle Cloud Infrastructure API signature. In the future, if Oracle modifies the method for signing requests, the version number will be incremented and your company will be notified.

Required Credentials and OCIDs

You need an API signing key in the correct format. See Required Keys and OCIDs.



Warning

Client Clock Skew

If the client's clock is skewed more than 5 minutes, a 401 (NotAuthenticated) HTTP status code is returned. This will affect your API requests. For more information, see Maximum Allowed Client Clock Skew.

You also need the OCIDs for your tenancy and user. See Where to Get the Tenancy's OCID and User's OCID.

Summary of Signing Steps

In general, these are the steps required to sign a request:

1. Form the HTTPS request (SSL protocol TLS 1.2 is required).
2. Create the signing string, which is based on parts of the request.
3. Create the signature from the signing string, using your private key and the RSA-SHA256 algorithm.
4. Add the resulting signature and other required information to the `Authorization` header in the request.

See the remaining sections in this topic for details about these steps.

Specification You Need to Be Familiar With

To learn how to perform steps 2-4 in the process above, refer to draft-cavage-http-signatures-08. It's a draft specification that forms the basis for how Oracle handles request signatures. It describes generally how to form the signing string, how to create the signature, and how to add the signature and required information to the request. The remaining sections in this topic assume you're familiar with it. Important details of the Oracle Cloud Infrastructure implementation of the reference are listed in the next section.

Special Implementation Details

The following sections describe important items to note about the Oracle Cloud Infrastructure implementation of the spec.

Authorization Header

The Oracle Cloud Infrastructure signature uses the "Signature" Authentication scheme (with an `Authorization` header), and not the Signature HTTP header.

Required Headers

This section describes the headers that must be included in the signing string.



Note

Error if Required Header is Missing

If a required header is missing, your client will receive a 401 "Unauthorized" response.

For GET and DELETE requests (when there's no content in the request body), the signing string must include at least these headers:

- `(request-target)` (as described in draft-cavage-http-signatures-08)
- `host`
- `date` or `x-date` (if both are included, Oracle uses `x-date`)

For PUT and POST requests (when there's content in the request body), the signing string must include at least these headers:

- `(request-target)`
- `host`
- `date` or `x-date` (if both are included, Oracle uses `x-date`)
- `x-content-sha256` (except for Object Storage PUT requests; see the next section)
- `content-type`
- `content-length`



Warning

For PUT and POST requests, your client must compute the `x-content-sha256` and include it in the request and signing string, even if the body is an empty string. Also, the `content-length` is always required in the request and signing string, even if the body is empty. Some HTTP clients will not send the `content-length` if the body is empty, so you must explicitly ensure your client sends it. If `date` and `x-date` are both included, Oracle uses `x-date`. The `x-date` is used to protect against the reuse of the signed portion of the request (replay attacks).

The one exception is for Object Storage PUT requests on objects (see the next section).

Special Instructions for Object Storage PUT

For Object Storage PutObject and UploadPart PUT requests, the signing string must include at least these headers:

- `(request-target)`
- `host`
- `date` or `x-date` (if both are included, Oracle uses `x-date`)

If the request also includes any of the other headers that are normally required for PUT requests (see the list above), then those headers must also be included in the signing string.

Case and Order of Headers

The headers must be all lowercase in the signing string.

The order of the headers in the signing string does not matter. Just make sure to specify the order in the `headers` parameter in the `Authorization` header, as described in the draft-cavage-http-signatures-05.



Warning

The `(request-target)` includes the path and query string from the request. Oracle expects that you will create the signing string with the query parameters in the *same order* as they appear in the request. If the request query parameters change order after signing occurs, authentication will fail.

URL Encoding of Path and Query String

When forming the signing string, you must URL encode all parameters in the path and query string (but not the headers) according to RFC 3986.

Key Identifier

You must set `keyId="<TENANCY OCID>/<USER OCID>/<KEY FINGERPRINT>"` in the `Authorization` header that you add to the request. To get those values, see Where to Get the Tenancy's OCID and User's OCID. An example `keyId` looks like this (wrapped to better fit the page):

```
ocid1.tenancy.oc1..exampleuwjnv47knr7uuuvqar5bshnspi6xoxsfebh3vy72fi4swgrkvuvq/ocid1.user.oc1..exampleb
a3pv6wkcr4jqae5f44n2b2m2yt2j6rx32uzr4h25vqstifsfdsq/40:a4:f8:a0:40:4f:a3:2f:e0:fd:4e:b9:25:72:81:5f
```

SIGNING ALGORITHM

The signing algorithm must be RSA-SHA256, and you must set `algorithm="rsa-sha256"` in the `Authorization` header (notice the quotation marks).

SIGNATURE VERSION

You should include `version="1"` in the `Authorization` header (notice the quotation marks). If you do not, it's assumed that you're using whatever the current version is (which is version 1 at this time).

EXAMPLE HEADER

Here's an example of the general syntax of the `Authorization` header (for a request with content in the body):

```
Authorization: Signature version="1",keyId="<tenancy_ocid>/<user_ocid>/<key_
fingerprint>",algorithm="rsa-sha256",headers="(request-target) date x-content-sha256 content-type
content-length",signature="Base64(RSA-SHA256(<signing_string>))"
```

Test Values

Here's an example key pair, two example requests, and the resulting `Authorization` header for each.



Warning

The example signatures use the RSA 2048-bit keys below. Use these keys only for testing your signing code, not for sending production requests.

```
-----BEGIN PUBLIC KEY-----
MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDCFENGw33yGihy92pDjZQhl0C3
6rPJj+CvfSC8+q28hxA161QFNUd13wuCTUcq0Qd2qsBe/2hFyc2DCJJg0h1L78+6
Z4UMR7EOcpfdUE9Hf3m/hs+FUR45uBJeDK1HSFHD8bHKD6kv8FPGfJTotc+2xjJw
oYi+1hqp1fIekaxsyQIDAQAB
```

```
-----END PUBLIC KEY-----

-----BEGIN RSA PRIVATE KEY-----
MIICXgIBAAKBgQDCFENGw33yGihy92pDjZQhl0C36rPJj+CvfSC8+q28hxA161QF
NUd13wuCTUcq0Qd2qsBe/2hFyc2DCJJg0h1L78+6Z4UMR7EOcpfdUE9Hf3m/hs+F
UR45uBJeDK1HSFHD8bHKD6kv8FPGfJTotc+2xjJwoYi+1hqp1fIekaxsyQIDAQAB
AoGBAJR8ZkCUvx5kzv+utdl7T5MnordT1TvoXXJGXK7ZZ+UuvMNUCdN2QPc4sBiA
QWvLw1cSKt5DsKZ8UETpYPy8pPYnnDEz2dDYiaew9+xEpubyeW2oH4Zx71wqBtOK
kqwrXa/pzdpiucRRjk6vE6YY7EBBs/g7uanVpGibOVAEsqH1AkEA7DkjVH28WDUg
f1nqvfn2Kj6CT7nIcE3jGJsZZ7zlZmBmHFDONMLUrXR/Zm3pR5m0tCmBqa5RK95u
412jt1dPIwJBANJT3v8pnkth48bQo/fKel6uEYyboRtA5/uHuHkZ6FQF7OUkGogc
mSJluOdc5t6hI1VsLn0QZEjQZMEOWr+wKSMCQQCC4kXJEsHAve77oP6HtG/IiEn7
kpyUXRNvFsDE0czpJJBvL/aRFUJxuRK91jhjC68sA7NsKMGg5OXb5I5Jj36xAkEA
gIT7aFOYBFwGgQAQkWNKLvySgKbAZRTeLBacpHMuQdl1DfdntvAyqpAZ0lY0RKmW
G6aFKaqQfOXKCyWoUiVknQJAXrlgySFci/2ueKlIE1QqIiLSZ8V8OlpFLRnb1pzI
7U1yQXnTAEFYM560yJlzUpOb1V4cScGd365tiSMvxLOvTA==
-----END RSA PRIVATE KEY-----


The public key is stored under keyId:


ocid1.tenancy.oc1..aaaaaaaaba3pv6wkcr4jqae5f15p2b2m2yt2j6rx32uzr4h25vqstifsfdsq/ocid1.user.oc1..aaaaaaa
at5nvwcna5j6aqzjcaty5eqbb6qt2jvpkanghtgdaqedqw3rynjq/73:61:a2:21:67:e0:df:be:7e:4b:93:1e:15:98:a5:b7



For the following GET request (line breaks inserted between query parameters for easier reading; also
notice the URL encoding as mentioned earlier):

GET https://iaas.us-phoenix-1.oraclecloud.com/20160918/instances
?availabilityDomain=Pjwf%3A%20PHX-AD-1
&compartmentId=ocid1.compartment.oc1..aaaaaaaam3we6vgnherjq5q2idnccdflvjsnog7mlr6rtdb25gilchfeyjxa
&displayName=TeamXInstances
&volumeId=ocid1.volume.oc1.phx.abyhqljrgvttnlx73nmrwfaux7kcvzfs3s66izvxf2h4lgvyndsdsnoiwr5q
Date: Thu, 05 Jan 2014 21:31:40 GMT

The signing string would be (line breaks inserted into the (request-target) header for easier reading):

date: Thu, 05 Jan 2014 21:31:40 GMT
(request-target): get /20160918/instances?availabilityDomain=Pjwf%3A%20PH
X-AD-1&compartmentId=ocid1.compartment.oc1..aaaaaaaam3we6vgnherjq5q2i
```

```
dnccdflvjsnog7mlr6rtdb25gilchfeyjxa&displayName=TeamXInstances&
volumeId=ocid1.volume.oc1.phx.abyhqljrgvttnlx73nmrwfaux7kcvzfs3s66izvxf2h
4lgvyndsdsnoiwr5q
host: iaas.us-phoenix-1.oraclecloud.com

The Authorization header would be:
Signature version="1",headers="date (request-target) host",keyId="ocid1.t
enancy.oc1..aaaaaaaaba3pv6wkcr4jqae5f15p2b2m2yt2j6rx32uzr4h25vqstifsfdsq/
ocid1.user.oc1..aaaaaaaat5nvwcna5j6aqzjcaty5eqbb6qt2jvpkanghtgdaqedqw3ryn
jq/73:61:a2:21:67:e0:df:be:7e:4b:93:1e:15:98:a5:b7",algorithm="rsa-sha256
",signature="GBas7grhyrhSKHP6AVIj/h5/Vp8bd/peM79H9Wv8kjoaCivujVXlpbKLjMPe
DUhxkFIWtTtLBj3sUzaFj34XE6YZAHc9r2DmE4pMwOAy/kiITcZxa1oHPOeRheC0jP2dqbTll
8fmTZVwKZOKHYPtrLJIJQHJjNvxFWeHQjMaR7M="

For the following POST request:

POST https://iaas.us-phoenix-1.oraclecloud.com/20160918/volumeAttachments
Date: Thu, 05 Jan 2014 21:31:40 GMT
{
   "compartmentId":
"ocid1.compartment.oc1..aaaaaaaam3we6vgnherjq5q2idnccdflvjsnog7mlr6rtdb25gilchfeyjxa",
   "instanceId": "ocid1.instance.oc1.phx.abuw4ljrlsfiqw6vzzxb43vyypt4pkodawglp3wqxjqofakrwvou52gb6s5a",
   "volumeId": "ocid1.volume.oc1.phx.abyhqljrgvttnlx73nmrwfaux7kcvzfs3s66izvxf2h4lgvyndsdsnoiwr5q"
}

The signing string would be:

date: Thu, 05 Jan 2014 21:31:40 GMT
(request-target): post /20160918/volumeAttachments
host: iaas.us-phoenix-1.oraclecloud.com
content-length: 316
content-type: application/json
x-content-sha256: V9Z20UJTvkvpJ50flBzKE32+6m2zJjweHpDMX/U4Uy0=

The Authorization header would be:

Signature version="1",headers="date (request-target) host content-length c
ontent-type x-content-sha256",keyId="ocid1.tenancy.oc1..aaaaaaaaba3pv6wkcr
4jqae5f15p2b2m2yt2j6rx32uzr4h25vqstifsfdsq/ocid1.user.oc1..aaaaaaaat5nvwcn
a5j6aqzjcaty5eqbb6qt2jvpkanghtgdaqedqw3rynjq/73:61:a2:21:67:e0:df:be:7e:4b
:93:1e:15:98:a5:b7",algorithm="rsa-sha256",signature="Mje8vIDPlwIHmD/cTDwR
xE7HaAvBg16JnVcsuqaNRim23fFPgQfLoOOxae6WqKb1uPjYEl0qIdazWaBy/Ml8DRhqlocMwo
```

```
SXv0fbukP8J5N80LCmzT/FFBvIvTB91XuXI3hYfP9Zt1l7S6ieVadHUfqBedWH0itrtPJBgKmrWso="
```

Sample Code

This section shows the basic code for signing API requests.

BASH

.

```
# Version: 1.0.2
# Usage:
# oci-curl <host> <method> [file-to-send-as-body] <request-target> [extra-curl-args]
#
# ex:
# oci-curl iaas.us-ashburn-1.oraclecloud.com get "/20160918/instances?compartmentId=some-compartment-
ocid"
# oci-curl iaas.us-ashburn-1.oraclecloud.com post ./request.json "/20160918/vcns"

function oci-curl {
       # TODO: update these values to your own
               local tenancyId="ocid1.tenancy.oc1..aaaaaaaaba3pv6wkcr4jqae5f15p2b2m2yt2j6rx32uzr4h25vqstifsfdsq";
               local authUserId="ocid1.user.oc1..aaaaaaaat5nvwcna5j6aqzjcaty5eqbb6qt2jvpkanghtgdaqedqw3rynjq";
               local keyFingerprint="20:3b:97:13:55:1c:5b:0d:d3:37:d8:50:4e:c5:3a:34";
               local privateKeyPath="/Users/someuser/.oci/oci_api_key.pem";

       local alg=rsa-sha256
       local sigVersion="1"
       local now="$(LC_ALL=C \date -u "+%a, %d %h %Y %H:%M:%S GMT")"
       local host=$1
       local method=$2
       local extra_args
       local keyId="$tenancyId/$authUserId/$keyFingerprint"

       case $method in

               "get" | "GET")
               local target=$3
               extra_args=("${@: 4}")
               local curl_method="GET";
               local request_method="get";
```

```
                ;;

                "delete" | "DELETE")
                local target=$3
                extra_args=("${@: 4}")
                local curl_method="DELETE";
                local request_method="delete";
                ;;

                "head" | "HEAD")
                local target=$3
                extra_args=("--head" "${@: 4}")
                local curl_method="HEAD";
                local request_method="head";
                ;;

                "post" | "POST")
                local body=$3
                local target=$4
                extra_args=("${@: 5}")
                local curl_method="POST";
                local request_method="post";
                local content_sha256="$(openssl dgst -binary -sha256 < $body | openssl enc -e -base64)";
                local content_type="application/json";
                local content_length="$(wc -c < $body | xargs)";
                ;;

                "put" | "PUT")
                local body=$3
                local target=$4
                extra_args=("${@: 5}")
                local curl_method="PUT"
                local request_method="put"
                local content_sha256="$(openssl dgst -binary -sha256 < $body | openssl enc -e -base64)";
                local content_type="application/json";
                local content_length="$(wc -c < $body | xargs)";
                ;;

                *) echo "invalid method"; return;;
esac

# This line will url encode all special characters in the request target except "/", "?", "=", and "&",
```

```
since those characters are used
# in the request target to indicate path and query string structure. If you need to encode any of "/",
"?", "=", or "&", such as when
# used as part of a path value or query string key or value, you will need to do that yourself in the
request target you pass in.

local escaped_target="$(echo $( rawurlencode "$target" ))"
local request_target="(request-target): $request_method $escaped_target"
local date_header="date: $now"
local host_header="host: $host"
local content_sha256_header="x-content-sha256: $content_sha256"
local content_type_header="content-type: $content_type"
local content_length_header="content-length: $content_length"
local signing_string="$request_target\n$date_header\n$host_header"
local headers="(request-target) date host"
local curl_header_args
curl_header_args=(-H "$date_header")
local body_arg
body_arg=()

if [ "$curl_method" = "PUT" -o "$curl_method" = "POST" ]; then
       signing_string="$signing_string\n$content_sha256_header\n$content_type_header\n$content_length_header"
       headers=$headers" x-content-sha256 content-type content-length"
       curl_header_args=("${curl_header_args[@]}" -H "$content_sha256_header" -H "$content_type_header" -H
"$content_length_header")
       body_arg=(--data-binary @${body})
fi

local sig=$(printf '%b' "$signing_string" | \
                      openssl dgst -sha256 -sign $privateKeyPath | \
                      openssl enc -e -base64 | tr -d '\n')

curl "${extra_args[@]}" "${body_arg[@]}" -X $curl_method -sS https://${host}${escaped_target} "${curl_
header_args[@]}" \
       -H "Authorization: Signature
version=\"$sigVersion\",keyId=\"$keyId\",algorithm=\"$alg\",headers=\"${headers}\",signature=\"$sig\""
}
# url encode all special characters except "/", "?", "=", and "&"
function rawurlencode {
  local string="${1}"
  local strlen=${#string}
  local encoded=""
```

```
  local pos c o

  for (( pos=0 ; pos<strlen ; pos++ )); do
       c=${string:$pos:1}
       case "$c" in
               [-_.~a-zA-Z0-9] | "/" | "?" | "=" | "&" ) o="${c}" ;;
               * )               printf -v o '%%%02x' "'$c"
       esac
       encoded+="${o}"
       done

       echo "${encoded}"
}
```

An example of a request.json file that could be used with the preceding Bash code is shown next:

```
{
   "compartmentId": "some-compartment-id",
   "displayName": "some-vcn-display-name",
   "cidrBlock": "10.0.0.0/16"
}
```

POWERSHELL

The following is an example for creating a request signature for an Oracle Cloud Infrastructure REST API call using a PowerShell script (`oci-rest.ps1`). The example uses the Bouncy Castle library .dll file to enable crypto functionality. Download the DLL from https://www.bouncycastle.org and place it in the same directory as the PowerShell script file.

.

```
param (
    [Parameter(Mandatory=$true)][string]$method,
    [Parameter(Mandatory=$true)][string]$ocihost,
    [Parameter(Mandatory=$true)][string]$target,

    [bool]$echo_debug = $false,

    # TODO: Update these defaults or override them on the command line.
    [string]$tenancyId =
'ocid1.tenancy.oc1..aaaaaaaaba3pv6wkcr4jqae5f15p2b2m2yt2j6rx32uzr4h25vqstifsfdsq',
    [string]$authUserId= 'ocid1.user.oc1..aaaaaaaalj3z3isgtuqd5uqft424he7r3cuqfr3e5gpidgnmqsxwd5qevkha',
```

```
    [string]$keyFingerprint = '29:f3:01:46:07:b8:dc:8c:16:c3:2b:b3:8d:dc:26:c5',
    [string]$privateKeyPath = $PSScriptRoot + '/oci_api_key.pem',

    [Parameter(Mandatory=$false)][string]$body,
    [Parameter(Mandatory=$false)][string]$bouncycastlelib
)

#############################################################################
# This is a powershell example of how to create request signatures for an
# Oracle Cloud Infra REST API call.  It was modeled after the bash example.
#
# Note that it utilizes the Bouncy Castle library dll for crypto functionality.
# It is assumed to be in the same directory as this script,
# but can be changed via commandline argument.
# See https://www.bouncycastle.org for more details.
#
# Usage:
# oci-rest.ps1 -host <host> -method <method> -body [file-to-send-as-body] -target <request-target> -
bouncycastlelib [BouncyCastle.Crypto.dll]
#
# Examples:
# ./oci-rest.ps1 -method get -ocihost iaas.us-ashburn-1.oraclecloud.com -target
"/20160918/instances?compartmentId=ocid1.compartment.oc1..aaaaaaaam3we6vgnherjq5q2idnccdflvjsnog7mlr6rt
db25gilchfeyjxa"
# ./oci-rest.ps1 -method post -ocihost iaas.us-ashburn-1.oraclecloud.com -target "/20160918/vcns" -body
./request.json
#
#############################################################################


#############################################################################
# Creates a message digest for the request body.
#############################################################################
function Digest($body_file_path) {
    $sha256digest = New-Object org.bouncycastle.crypto.digests.SHA256Digest
    $content = Get-Item $body_file_path
    $bytes = $null
    if ($PSVersionTable.PSVersion.Major -ge 6) {
        [byte[]]$bytes = Get-Content $body_file_path -AsByteStream
    } else {
        [byte[]]$bytes = Get-Content $body_file_path -Encoding byte
    }
```

```
    $sha256digest.BlockUpdate($bytes, 0, $bytes.Length)
    $result_size = $sha256digest.GetDigestSize()
    $result_bytes = New-Object Byte[] $result_size
    $sha256digest.DoFinal($result_bytes, 0) | Out-Null
    $content_sha256 = [Convert]::ToBase64String($result_bytes)
    return $content_sha256
}

################################################################################
# Creates the signature to be put in the Authorization request header.
################################################################################
function Sign($signing_string, $privateKeyPath) {
    $sha256digest = New-Object org.bouncycastle.crypto.digests.SHA256Digest
    $signer = New-Object Org.BouncyCastle.Crypto.Signers.RSADigestSigner $sha256digest

    $privateKeyFile = [System.IO.File]::OpenText($privateKeyPath)
    $pemReader = New-Object Org.BouncyCastle.OpenSsl.PemReader $privateKeyFile
    $keyPair = [Org.BouncyCastle.Crypto.AsymmetricCipherKeyPair]$pemReader.ReadObject()
    #$keyParameter = [Org.BouncyCastle.Security.DotNetUtilities]::ToRSAParameters($keyPair.Private)
    $keyParameter = $keyPair.Private
    $signer.Init($true, $keyParameter)

    $encoding = [System.Text.Encoding]::UTF8
    [byte[]]$bytes = $encoding.GetBytes($signing_string)
    $signer.BlockUpdate($bytes, 0, $bytes.Length)
    $signature = $signer.GenerateSignature()
    $signedString = [Convert]::ToBase64String($signature)

    return $signedString
}

################################################################################
# Makes the Oracle Cloud API REST call.
################################################################################
function RestCall($method, $ocihost, $target, $privateKeyPath, $keyId,
                  $body_file_path='', $echo_debug=$false) {
    $alg = 'rsa-sha256'
    $sigVersion = '1'
    $now = Get-Date
    $now = $now.ToUniversalTime()
    $now_string = $now.ToString("ddd, dd MMM yyyy HH:mm:ss") + " GMT"
```

```
      $content_type = ''
      $content_length = 0
      $content_sha256 = ''
   $request_method = $method.ToLower()
   If ($request_method -eq "get") {
       $method = "Get"
   } ElseIf ($request_method -eq "delete") {
       $method = "Delete"
   } ElseIf ($request_method -eq "head") {
       $method = "Head"
   } ElseIf ($request_method -eq "post") {
       if ($body_file_path.Length -eq 0) {
           echo "body parameter must be specified and point to valid json body file."
           Exit 1
       }
       $method = "Post"
              $content_type = 'application/json'
              $content_length = (Get-Item $body_file_path).length
       $content_sha256 = Digest $body_file_path
       if ($echo_debug) {
           output_debug "digest=$content_sha256"
       }
   } ElseIf ($request_method -eq "put") {
       if ($body_file_path.Length -eq 0) {
           echo "body parameter must be specified and point to valid json body file."
           Exit 1
       }
       $method = "Put"
              $content_type = 'application/json'
              $content_length = (Get-Item $body_file_path).length
       $content_sha256 = Digest $body_file_path
       if ($echo_debug) {
           output_debug "digest=$content_sha256"
       }
   } Else {
       echo "invalid method"
       Exit 1
      }

   $escaped_target = rawurlencode $target
   $request_target = $request_method + " " + $escaped_target
   if ($echo_debug) {
```

```
        output_debug "escaped target=$escaped_target"
    }

    $headers = @{}
    $header_list = "(request-target) date host"
    $headers["date"] = $now_string
    $headers["host"] = $ocihost

    #$nl = [Environment]::NewLine   # This doesn't work in windows environments
    $nl = "`n"
       $signing_string = "(request-target): " + $request_target + $nl + "date: " + $now_string + $nl + "host:
" + $ocihost

    if (($request_method -eq "put") -or ($request_method -eq "post")) {
               $headers["x-content-sha256"] = $content_sha256
        $headers["content-type"] = $content_type
        $headers["content-length"] = $content_length
               $header_list = $header_list + " x-content-sha256 content-type content-length"
        $signing_string = $signing_string + $nl + "x-content-sha256: " + $content_sha256 + $nl +
                          "content-type: " + $content_type + $nl + "content-length: " + $content_length
       }

    if ($echo_debug) {
        output_debug "signing string=$signing_string"
    }
    $sig = Sign $signing_string $privateKeyPath
    if ($echo_debug) {
        output_debug "signature=$sig"
    }
    $authorization = 'Signature version="' + $sigVersion + '",keyId="' + $keyId + '",algorithm="' +
                     $alg + '",headers="' + $header_list + '",signature="' + $sig + '"'
    $headers["Authorization"] = $authorization

    $url = "https://" + $ocihost + $escaped_target
    if ($echo_debug) {
        output_debug "authorization=$authorization"
        output_debug "url=$url"
        output_debug "headers=$headers"
        $headers.getenumerator() | Out-String
    }

    # Without this setting, Invoke-RestMethod was failing on windows with a connection error.
```

```
    [Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12

       if ($body_file_path.Length -gt 0) {
        PostPutRequest $method $url $headers $body_file_path
       } else {
        Invoke-RestMethod -Method $method -Uri $url -Headers $headers | ConvertTo-Json
       }
}

################################################################################
# Makes a Post or Put call.
# We do this b/c Invoke-RestMethod doesn't seem to give the granular control
# needed, but HttpWebRequest works well.
################################################################################
Function PostPutRequest($method, $url, $headers, $body_file_path) {
    $junk = [System.Reflection.Assembly]::LoadWithPartialName("System.Web")
       $request = [System.Net.HttpWebRequest]::Create($url)
    $request.Method = $method.ToUpper()
    $request.Accept = "application/json";
       $request.ProtocolVersion = "1.0"
    $request.ContentType = $headers["content-type"]
    $request.ContentLength = $headers["content-length"]
    $request.Date = $headers["date"]
    $request.Host = $headers["host"]
    $request.Headers["x-content-sha256"] = $headers["x-content-sha256"]
    $request.Headers["authorization"] = $headers["authorization"]
    #$request.Headers["(request-target)"] = $headers["(request-target)"]

    # Create the input stream to the REST API
       $requestInputStream = $request.GetRequestStream()

       # Create a stream writer to write the json
       $writer = New-Object System.IO.StreamWriter($requestInputStream)
       $writer.AutoFlush = $true

       # Write the json
       Try {
        $bytes = $null
        if ($PSVersionTable.PSVersion.Major -ge 6) {
            [byte[]]$bytes = Get-Content $body_file_path -AsByteStream
        } else {
            [byte[]]$bytes = Get-Content $body_file_path -Encoding byte
```

```
        }
                $writer.Write($bytes, 0, $bytes.Length)
        } Catch [System.IO.IOException] {
                Throw "Cannot write to stream. Exception $($_.Exception)"
        } Catch [System.Exception] {
                Throw "Some other weird error caught...$($_.Exception)"
        } Finally {
                $writer.Close()
        }
        Get-WebResponseOutput $request
}

##############################################################################
# Gets the response output for a request.
##############################################################################
Function Get-WebResponseOutput($request) {
        $junk = [System.Reflection.Assembly]::LoadWithPartialName("System.Web")

    $response = $null
        Try {
                $response = $request.GetResponse()
        } Catch [System.Net.WebException] {
         echo "Exception from server: " $_.Exception.Message
         $ex = $_.Exception.Response.StatusCode
         if ($response -ne $null) {
             $response.Close()
         }
                Throw "Exception from server: $ex"
        } Catch [System.Exception] {
         if ($response -ne $null) {
             $response.Close()
         }
         echo "Some other random error: " $_.Exception.Message
                Throw "Some other random error..$($_.Exception)"
        }

        $readStream = $response.GetResponseStream()
        $reader = New-Object System.IO.StreamReader($readStream)

        Try {
                $output = $reader.readtoend()
        } Catch {
```

```
        echo "Exception reading stream from server.  Exception: " $_.Exception.Message
               Throw "Exception reading stream from server.  Exception: $($_.Exception)"
       } Finally {
        if ($response -ne $null) {
            $response.Close()
        }
               $reader.Close()
       }
       return $output
}


################################################################################
# url encode all special characters except "/", "?", "=", and "&"
# This will url encode all special characters in the request target except "/", "?", "=", and "&",
# since those characters are used in the request target to indicate path and query string structure.
# If you need to encode any of "/", "?", "=", or "&", such as, when used
# as part of a path value or query string key or value,
# you will need to do that yourself in the request target you pass in.
################################################################################
function rawurlencode($target) {
       Add-Type -AssemblyName System.Web
       $chars_to_skip = "-_.~abcdefghijklmnopqrstuvwxzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789/?=&";
    $encoded = ""
       for ($i=0; $i -lt $target.Length; $i++) {
               $ch = $target[$i]
               $o = $ch
               if ($chars_to_skip.IndexOf($ch) -lt 0) {
            $o = [System.Web.HttpUtility]::UrlEncode($ch)
               }
               $encoded = $encoded + $o
    }
    return $encoded
}


################################################################################
# Trivial function to output debug messages to console.
################################################################################
function output_debug($msg) {
    echo "[debug] $msg"
    #Write-Verbose $msg
```

```
}

################################################################################
# Main entry point logic.
################################################################################
if ($bouncycastlelib.Length -eq 0) {
    $bouncycastlelib = $PSScriptRoot + "/BouncyCastle.Crypto.dll"
}
if ($echo_debug) {
    [Reflection.Assembly]::LoadFile($bouncycastlelib)
    output_debug "Bouncy Castle loaded and ready"
} else {
    [Reflection.Assembly]::LoadFile($bouncycastlelib) | Out-Null
}

$keyId = $tenancyId + "/" + $authUserId + "/" + $keyFingerprint
if ($echo_debug) {
    output_debug "keyId=$keyId"
}

RestCall $method $ocihost $target $privateKeyPath $keyId $body $echo_debug
```

Following is an example of a `request.json` file that you can use with the preceding PowerShell code:

```
{
    "compartmentId": "some-compartment-id",
    "displayName": "some-vcn-display-name",
    "cidrBlock": "10.0.0.0/16"
}
```

C#

.

```
// Version 1.0.1
namespace Oracle.Oci
{
    using System;
    using System.Collections.Generic;
    using System.IO;
```

```
    using System.Net;
    using System.Security.Cryptography;
    using System.Text;

    //
    //  Nuget Package Manager Console: Install-Package BouncyCastle
    //  Nuget CLI: nuget install BouncyCastle
    //
    using Org.BouncyCastle.Crypto;
    using Org.BouncyCastle.Crypto.Parameters;
    using Org.BouncyCastle.OpenSsl;
    using Org.BouncyCastle.Security;

    public class Signing
    {
        public static void Main(string[] args)
        {
            var tenancyId =
"ocid1.tenancy.oc1..aaaaaaaaba3pv6wkcr4jqae5f15p2b2m2yt2j6rx32uzr4h25vqstifsfdsq";
            var compartmentId =
"ocid1.compartment.oc1..aaaaaaaam3we6vgnherjq5q2idnccdflvjsnog7mlr6rtdb25gilchfeyjxa";
            var userId = "ocid1.user.oc1..aaaaaaaat5nvwcna5j6aqzjcaty5eqbb6qt2jvpkanghtgdaqedqw3rynjq";
            var fingerprint = "20:3b:97:13:55:1c:5b:0d:d3:37:d8:50:4e:c5:3a:34";
            var privateKeyPath = "private.pem";
            var privateKeyPassphrase = "password";

            var signer = new RequestSigner(tenancyId, userId, fingerprint, privateKeyPath,
privateKeyPassphrase);

            // Oracle Cloud Infrastructure APIs require TLS 1.2
            // uncomment the line below if targeting < .NET Framework 4.6 (HttpWebRequest does not
enable TLS 1.2 by default in earlier versions)
            // ServicePointManager.SecurityProtocol = SecurityProtocolType.Tls12;

            // GET with query parameters (gets user)
            var uri = new Uri($"https://identity.us-phoenix-1.oraclecloud.com/20160918/users/{userId}");
            var request = (HttpWebRequest)WebRequest.Create(uri);
            request.Method = "GET";
            request.Accept = "application/json";

            signer.SignRequest(request);
```

```
            Console.WriteLine($"Authorization header: {request.Headers["authorization"]}");

            ExecuteRequest(request);

            // POST with body (creates a VCN)
            uri = new Uri($"https://iaas.us-phoenix-1.oraclecloud.com/20160918/vcns");
            var body = string.Format(@"{{""cidrBlock"" : ""10.0.0.0/16"",""compartmentId"" : ""
{0}"",""displayName"" : ""MyVcn""}}", compartmentId);
            var bytes = Encoding.UTF8.GetBytes(body);

            request = (HttpWebRequest)WebRequest.Create(uri);
            request.Method = "POST";
            request.Accept = "application/json";
            request.ContentType = "application/json";
            request.Headers["x-content-sha256"] = Convert.ToBase64String(SHA256.Create().ComputeHash
(bytes));

            using (var stream = request.GetRequestStream())
            {
                stream.Write(bytes, 0, bytes.Length);
            }

            signer.SignRequest(request);

            Console.WriteLine($"Authorization header: {request.Headers["authorization"]}");

            ExecuteRequest(request);



            // GET with query parameters (gets namespace)
            uri = new Uri($"https://objectstorage.us-phoenix-1.oraclecloud.com/n/");
            request = (HttpWebRequest)WebRequest.Create(uri);
            request.Method = "GET";
            request.Accept = "application/json";

            signer.SignRequest(request);

            Console.WriteLine($"Authorization header: {request.Headers["authorization"]}");

            string namespaceName = ExecuteRequest(request);
```

```
            namespaceName = JsonConvert.DeserializeObject<string>(namespaceName);

            // POST with body (creates a bucket)
            uri = new Uri($"https://objectstorage.us-phoenix-1.oraclecloud.com/n/{namespaceName}/b/" );
            body = string.Format(@"{{""name"" : ""bucket01"",""compartmentId"" : ""
{0}"",""publicAccessType"" : ""ObjectRead""}}", compartmentId);
            bytes = Encoding.UTF8.GetBytes(body);

            request = (HttpWebRequest)WebRequest.Create(uri);
            request.Method = "POST";
            request.Accept = "application/json";
            request.ContentType = "application/json";
            request.Headers["x-content-sha256"] = Convert.ToBase64String(SHA256.Create().ComputeHash
(bytes));

            using (var stream = request.GetRequestStream())
            {
                stream.Write(bytes, 0, bytes.Length);
            }

            signer.SignRequest(request);

            Console.WriteLine($"Authorization header: {request.Headers["authorization"]}");

            ExecuteRequest(request);


            // PUT with body (puts a object)
            uri = new Uri($"https://objectstorage.us-phoenix-1.oraclecloud.com/n/
{namespaceName}/b/bucket01/o/object01");
            body = "Hello Object Storage Service!!!";
            bytes = Encoding.UTF8.GetBytes(body);

            request = (HttpWebRequest)WebRequest.Create(uri);
            request.Method = "PUT";
            request.Accept = "application/json";
            request.ContentType = "application/json";

            using (var stream = request.GetRequestStream())
            {
                stream.Write(bytes, 0, bytes.Length);
            }
```

```
            signer.SignRequest(request, true);

            Console.WriteLine($"Authorization header: {request.Headers["authorization"]}");

            ExecuteRequest(request);

            // POST with body (create multipart upload)
            uri = new Uri($"https://objectstorage.us-phoenix-1.oraclecloud.com/n/
{namespaceName}/b/bucket01/u");
            body = "{\"object\" : \"object02\"}";
            bytes = Encoding.UTF8.GetBytes(body);

            request = (HttpWebRequest)WebRequest.Create(uri);
            request.Method = "POST";
            request.Accept = "application/json";
            request.ContentType = "application/json";
            request.Headers["x-content-sha256"] = Convert.ToBase64String(SHA256.Create().ComputeHash
(bytes));

            using (var stream = request.GetRequestStream())
            {
                stream.Write(bytes, 0, bytes.Length);
            }

            signer.SignRequest(request);

            Console.WriteLine($"Authorization header: {request.Headers["authorization"]}");

            ExecuteRequest(request);

            Console.ReadKey();
        }

        private static string ExecuteRequest(HttpWebRequest request)
        {
            try
            {
                var webResponse = (HttpWebResponse)request.GetResponse();
                var response = new StreamReader(webResponse.GetResponseStream()).ReadToEnd();
                Console.WriteLine($"Response: {response}");
```

```
                return response;
            }
            catch (WebException e)
            {
                Console.WriteLine($"Exception occurred: {e.Message}");
                Console.WriteLine($"Response: {new StreamReader(e.Response.GetResponseStream
()).ReadToEnd()}");

                return String.Empty;
            }
        }

        public class RequestSigner
        {
            private static readonly IDictionary<string, List<string>> RequiredHeaders = new
Dictionary<string, List<string>>
            {
                { "GET", new List<string>{"date", "(request-target)", "host" }},
                { "HEAD", new List<string>{"date", "(request-target)", "host" }},
                { "DELETE", new List<string>{"date", "(request-target)", "host" }},
                { "PUT", new List<string>{"date", "(request-target)", "host", "content-length",
"content-type", "x-content-sha256" }},
                { "POST", new List<string>{"date", "(request-target)", "host", "content-length",
"content-type", "x-content-sha256" }},
                { "PUT-LESS", new List<string>{"date", "(request-target)", "host" }}
            };

            private readonly string keyId;
            private readonly ISigner signer;

            /// <summary>
            /// Adds the necessary authorization header for signed requests to Oracle Cloud
Infrastructure services.
            /// Documentation for request signatures can be found here:
https://docs.cloud.oracle.com/Content/API/Concepts/signingrequests.htm
            /// </summary>
            /// <param name="tenancyId">The tenancy OCID</param>
            /// <param name="userId">The user OCID</param>
            /// <param name="fingerprint">The fingerprint corresponding to the provided key</param>
            /// <param name="privateKeyPath">Path to a PEM file containing a private key</param>
            /// <param name="privateKeyPassphrase">An optional passphrase for the private key</param>
            public RequestSigner(string tenancyId, string userId, string fingerprint, string
```

```
privateKeyPath, string privateKeyPassphrase="")
            {
                // This is the keyId for a key uploaded through the console
                this.keyId = $"{tenancyId}/{userId}/{fingerprint}";

                AsymmetricCipherKeyPair keyPair;
                using (var fileStream = File.OpenText(privateKeyPath))
                {
                    try {
                        keyPair = (AsymmetricCipherKeyPair)new PemReader(fileStream, new Password
(privateKeyPassphrase.ToCharArray())).ReadObject();
                    }
                    catch (InvalidCipherTextException) {
                        throw new ArgumentException("Incorrect passphrase for private key");
                    }
                }

                RsaKeyParameters privateKeyParams = (RsaKeyParameters)keyPair.Private;
                this.signer = SignerUtilities.GetSigner("SHA-256withRSA");
                this.signer.Init(true, privateKeyParams);
            }

            public void SignRequest(HttpWebRequest request, bool useLessHeadersForPut = false)
            {
                if (request == null) { throw new ArgumentNullException(nameof(request)); }

                // By default, request.Date is DateTime.MinValue, so override to DateTime.UtcNow, but
preserve the value if caller has already set the Date
                if (request.Date == DateTime.MinValue) { request.Date = DateTime.UtcNow; }

                var requestMethodUpper = request.Method.ToUpperInvariant();
                var requestMethodKey = useLessHeadersForPut ? requestMethodUpper + "-LESS" :
requestMethodUpper;

                List<string> headers;
                if (!RequiredHeaders.TryGetValue(requestMethodKey, out headers)) {
                    throw new ArgumentException($"Don't know how to sign method: {request.Method}");
                }

                // for PUT and POST, if the body is empty we still must explicitly set content-length =
0 and x-content-sha256
                // the caller may already do this, but we shouldn't require it since we can determine it
```

```
here
                if (request.ContentLength <= 0 && (string.Equals(requestMethodUpper, "POST") ||
string.Equals(requestMethodUpper, "PUT")))
                {
                    request.ContentLength = 0;
                    request.Headers["x-content-sha256"] = Convert.ToBase64String(SHA256.Create
().ComputeHash(new byte[0]));
                }

                var signingStringBuilder = new StringBuilder();
                var newline = string.Empty;
                foreach (var headerName in headers)
                {
                    string value = null;
                    switch (headerName)
                    {
                        case "(request-target)":
                            value = buildRequestTarget(request);
                            break;
                        case "host":
                            value = request.Host;
                            break;
                        case "content-length":
                            value = request.ContentLength.ToString();
                            break;
                        default:
                            value = request.Headers[headerName];
                            break;
                    }

                    if (value == null) { throw new ArgumentException($"Request did not contain required
header: {headerName}"); }
                    signingStringBuilder.Append(newline).Append($"{headerName}: {value}");
                    newline = "\n";
                }

                // generate signature using the private key
                var bytes = Encoding.UTF8.GetBytes(signingStringBuilder.ToString());
                this.signer.BlockUpdate(bytes, 0, bytes.Length);
                var signature = Convert.ToBase64String(this.signer.GenerateSignature());
                var authorization = $@"Signature version=""1"",headers=""{string.Join(" ",
headers)}"",keyId=""{keyId}"",algorithm=""rsa-sha256"",signature=""{signature}""";
```

```
                request.Headers["authorization"] = authorization;
            }

            private static string buildRequestTarget(HttpWebRequest request)
            {
                // ex. get /20160918/instances
                return $"{request.Method.ToLowerInvariant()} {request.RequestUri.PathAndQuery}";
            }
        }

        /// <summary>
        /// Implements Bouncy Castle's IPasswordFinder interface to allow opening password protected
private keys.
        /// </summary>
        public class Password : IPasswordFinder
        {
            private readonly char[] password;

            public Password(char[] password) { this.password = password; }

            public char[] GetPassword() { return (char[])password.Clone(); }
        }
    }
}
```

JAVA

This sample omits the optional `version` field in the Authorization header.

```
/*
* @version 1.0.1
*
<dependency>
    <groupId>com.google.guava</groupId>
    <artifactId>guava</artifactId>
    <version>19.0</version>
</dependency>
 */
import com.google.common.collect.ImmutableList;
import com.google.common.collect.ImmutableMap;
import com.google.common.hash.Hashing;
/*
<dependency>
```

```
    <groupId>org.apache.httpcomponents</groupId>
    <artifactId>httpclient</artifactId>
    <version>4.5</version>
</dependency>
*/
import org.apache.http.HttpEntity;
import org.apache.http.client.methods.HttpEntityEnclosingRequestBase;
import org.apache.http.client.methods.HttpGet;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.client.methods.HttpRequestBase;
import org.apache.http.entity.ByteArrayEntity;
/*
<dependency>
    <groupId>org.tomitribe</groupId>
    <artifactId>tomitribe-http-signatures</artifactId>
    <version>1.0</version>
</dependency>
*/
import org.apache.http.entity.StringEntity;
import org.tomitribe.auth.signatures.MissingRequiredHeaderException;
import org.tomitribe.auth.signatures.PEM;
import org.tomitribe.auth.signatures.Signature;
import org.tomitribe.auth.signatures.Signer;


import java.io.ByteArrayOutputStream;
import java.io.IOException;
import java.io.InputStream;
import java.io.UnsupportedEncodingException;
import java.net.URI;
import java.nio.charset.StandardCharsets;
import java.nio.file.Files;
import java.nio.file.Paths;
import java.security.Key;
import java.security.PrivateKey;
import java.security.spec.InvalidKeySpecException;
import java.text.SimpleDateFormat;
import java.util.*;
import java.util.stream.Collectors;

/**
 * This example creates a {@link RequestSigner}, then prints out the Authorization header
```

```
 * that is inserted into the HttpGet object.
 *
 * <p>
 * apiKey is the identifier for a key uploaded through the console.
 * privateKeyFilename is the location of your private key (that matches the uploaded public key for
apiKey).
 * </p>
 *
 * The signed HttpGet request is not executed, since instanceId does not map to a real instance.
 */
public class Signing {
    public static void main(String[] args) throws UnsupportedEncodingException {
        HttpRequestBase request;

        // This is the keyId for a key uploaded through the console
        String apiKey =
("ocid1.tenancy.oc1..aaaaaaaaba3pv6wkcr4jqae5f15p2b2m2yt2j6rx32uzr4h25vqstifsfdsq/"
                                +
"ocid1.user.oc1..aaaaaaaat5nvwcna5j6aqzjcaty5eqbb6qt2jvpkanghtgdaqedqw3rynjq/"
                                + "20:3b:97:13:55:1c:5b:0d:d3:37:d8:50:4e:c5:3a:34");
        String privateKeyFilename = "../sample-private-key";
        PrivateKey privateKey = loadPrivateKey(privateKeyFilename);
        RequestSigner signer = new RequestSigner(apiKey, privateKey);

        // GET with query parameters
        String uri = "https://iaas.us-ashburn-
1.oraclecloud.com/20160918/instances?availabilityDomain=%s&compartmentId=%s&displayName=%s&volumeId=%s";
        uri = String.format(uri,
                "Pjwf%3A%20PHX-AD-1",
                // Older ocid formats included ":" which must be escaped

"ocid1.compartment.oc1..aaaaaaaam3we6vgnherjq5q2idnccdflvjsnog7mlr6rtdb25gilchfeyjxa".replace(":",
"%3A"),
                "TeamXInstances",

"ocid1.volume.oc1.phx.abyhqljrgvttnlx73nmrwfaux7kcvzfs3s66izvxf2h4lgvyndsdsnoiwr5q".replace(":", "%3A")
        );

        request = new HttpGet(uri);
        // Uncomment to use a fixed date
        // request.setHeader("Date", "Thu, 05 Jan 2014 21:31:40 GMT");
```

```
        signer.signRequest(request);
        System.out.println(uri);
        System.out.println(request.getFirstHeader("Authorization"));


        // POST with body
        uri = "https://iaas.us-ashburn-1.oraclecloud.com/20160918/volumeAttachments";
        request = new HttpPost(uri);
        // Uncomment to use a fixed date
        // request.setHeader("Date", "Thu, 05 Jan 2014 21:31:40 GMT");
        HttpEntity entity = new StringEntity("{\n" +
                "    \"compartmentId\":
\"ocid1.compartment.oc1..aaaaaaaam3we6vgnherjq5q2idnccdflvjsnog7mlr6rtdb25gilchfeyjxa\",\n" +
                "    \"instanceId\":
\"ocid1.instance.oc1.phx.abuw4ljrlsfiqw6vzzxb43vyypt4pkodawglp3wqxjqofakrwvou52gb6s5a\",\n" +
                "    \"volumeId\":
\"ocid1.volume.oc1.phx.abyhqljrgvttnlx73nmrwfaux7kcvzfs3s66izvxf2h4lgvyndsdsnoiwr5q\"\n" +
                "}");
        ((HttpPost)request).setEntity(entity);

        signer.signRequest(request);
        System.out.println("\n" + uri);
        System.out.println(request.getFirstHeader("Authorization"));

    }

    /**
     * Load a {@link PrivateKey} from a file.
     */
    private static PrivateKey loadPrivateKey(String privateKeyFilename) {
        try (InputStream privateKeyStream = Files.newInputStream(Paths.get(privateKeyFilename))){
            return PEM.readPrivateKey(privateKeyStream);
        } catch (InvalidKeySpecException e) {
                throw new RuntimeException("Invalid format for private key");
        } catch (IOException e) {
            throw new RuntimeException("Failed to load private key");
        }
    }

    /**
     * A light wrapper around https://github.com/tomitribe/http-signatures-java
     */
```

```
    public static class RequestSigner {
        private static final SimpleDateFormat DATE_FORMAT;
        private static final String SIGNATURE_ALGORITHM = "rsa-sha256";
        private static final Map<String, List<String>> REQUIRED_HEADERS;
        static {
            DATE_FORMAT = new SimpleDateFormat("EEE, dd MMM yyyy HH:mm:ss zzz", Locale.US);
            DATE_FORMAT.setTimeZone(TimeZone.getTimeZone("GMT"));
            REQUIRED_HEADERS = ImmutableMap.<String, List<String>>builder()
                    .put("get", ImmutableList.of("date", "(request-target)", "host"))
                    .put("head", ImmutableList.of("date", "(request-target)", "host"))
                    .put("delete", ImmutableList.of("date", "(request-target)", "host"))
                    .put("put", ImmutableList.of("date", "(request-target)", "host", "content-length",
"content-type", "x-content-sha256"))
                    .put("post", ImmutableList.of("date", "(request-target)", "host", "content-length",
"content-type", "x-content-sha256"))
            .build();
        }
        private final Map<String, Signer> signers;

        /**
         * @param apiKey The identifier for a key uploaded through the console.
         * @param privateKey The private key that matches the uploaded public key for the given apiKey.
         */
        public RequestSigner(String apiKey, Key privateKey) {
            this.signers = REQUIRED_HEADERS
                    .entrySet().stream()
                    .collect(Collectors.toMap(
                            entry -> entry.getKey(),
                            entry -> buildSigner(apiKey, privateKey, entry.getKey())));
        }

        /**
         * Create a {@link Signer} that expects the headers for a given method.
         * @param apiKey The identifier for a key uploaded through the console.
         * @param privateKey The private key that matches the uploaded public key for the given apiKey.
         * @param method HTTP verb for this signer
         * @return
         */
        protected Signer buildSigner(String apiKey, Key privateKey, String method) {
            final Signature signature = new Signature(
                    apiKey, SIGNATURE_ALGORITHM, null, REQUIRED_HEADERS.get(method.toLowerCase()));
            return new Signer(privateKey, signature);
```

```
        }

        /**
         * Sign a request, optionally including additional headers in the signature.
         *
         * <ol>
         * <li>If missing, insert the Date header (RFC 2822).</li>
         * <li>If PUT or POST, insert any missing content-type, content-length, x-content-sha256</li>
         * <li>Verify that all headers to be signed are present.</li>
         * <li>Set the request's Authorization header to the computed signature.</li>
         * </ol>
         *
         * @param request The request to sign
         */
        public void signRequest(HttpRequestBase request) {
            final String method = request.getMethod().toLowerCase();
            // nothing to sign for options
            if (method.equals("options")) {
                return;
            }

            final String path = extractPath(request.getURI());

            // supply date if missing
            if (!request.containsHeader("date")) {
                request.addHeader("date", DATE_FORMAT.format(new Date()));
            }

            // supply host if mossing
            if (!request.containsHeader("host")) {
                request.addHeader("host", request.getURI().getHost());
            }

            // supply content-type, content-length, and x-content-sha256 if missing (PUT and POST only)
            if (method.equals("put") || method.equals("post")) {
                if (!request.containsHeader("content-type")) {
                    request.addHeader("content-type", "application/json");
                }
                if (!request.containsHeader("content-length") || !request.containsHeader("x-content-
sha256")) {
                    byte[] body = getRequestBody((HttpEntityEnclosingRequestBase) request);
                    if (!request.containsHeader("content-length")) {
```

```
                    request.addHeader("content-length", Integer.toString(body.length));
                }
                if (!request.containsHeader("x-content-sha256")) {
                    request.addHeader("x-content-sha256", calculateSHA256(body));
                }
            }
        }

        final Map<String, String> headers = extractHeadersToSign(request);
        final String signature = this.calculateSignature(method, path, headers);
        request.setHeader("Authorization", signature);
    }

    /**
     * Extract path and query string to build the (request-target) pseudo-header.
     * For the URI "http://www.host.com/somePath?example=path" return "/somePath?example=path"
     */
    private static String extractPath(URI uri) {
        String path = uri.getRawPath();
        String query = uri.getRawQuery();
        if (query != null && !query.trim().isEmpty()) {
            path = path + "?" + query;
        }
        return path;
    }

    /**
     * Extract the headers required for signing from a {@link HttpRequestBase}, into a Map
     * that can be passed to {@link RequestSigner#calculateSignature}.
     *
     * <p>
     * Throws if a required header is missing, or if there are multiple values for a single header.
     * </p>
     *
     * @param request The request to extract headers from.
     */
    private static Map<String, String> extractHeadersToSign(HttpRequestBase request) {
        List<String> headersToSign = REQUIRED_HEADERS.get(request.getMethod().toLowerCase());
        if (headersToSign == null) {
            throw new RuntimeException("Don't know how to sign method " + request.getMethod());
        }
        return headersToSign.stream()
```

```
                // (request-target) is a pseudo-header
                .filter(header -> !header.toLowerCase().equals("(request-target)"))
                .collect(Collectors.toMap(
                header -> header,
                header -> {
                    if (!request.containsHeader(header)) {
                        throw new MissingRequiredHeaderException(header);
                    }
                    if (request.getHeaders(header).length > 1) {
                        throw new RuntimeException(
                                String.format("Expected one value for header %s", header));
                    }
                    return request.getFirstHeader(header).getValue();
                }));
    }

    /**
     * Wrapper around {@link Signer#sign}, returns the {@link Signature} as a String.
     *
     * @param method Request method (GET, POST, ...)
     * @param path The path + query string for forming the (request-target) pseudo-header
     * @param headers Headers to include in the signature.
     */
    private String calculateSignature(String method, String path, Map<String, String> headers) {
        Signer signer = this.signers.get(method);
        if (signer == null) {
            throw new RuntimeException("Don't know how to sign method " + method);
        }
        try {
            return signer.sign(method, path, headers).toString();
        } catch (IOException e) {
            throw new RuntimeException("Failed to generate signature", e);
        }
    }

    /**
     * Calculate the Base64-encoded string representing the SHA256 of a request body
     * @param body The request body to hash
     */
    private String calculateSHA256(byte[] body) {
        byte[] hash = Hashing.sha256().hashBytes(body).asBytes();
        return Base64.getEncoder().encodeToString(hash);
```

```
        }

        /**
         * Helper to safely extract a request body.  Because an {@link HttpEntity} may not be
repeatable,
         * this function ensures the entity is reset after reading.  Null entities are treated as an
empty string.
         *
         * @param request A request with a (possibly null) {@link HttpEntity}
         */
        private byte[] getRequestBody(HttpEntityEnclosingRequestBase request) {
            HttpEntity entity = request.getEntity();
            // null body is equivalent to an empty string
            if (entity == null) {
                return "".getBytes(StandardCharsets.UTF_8);
            }
            // May need to replace the request entity after consuming
            boolean consumed = !entity.isRepeatable();
            ByteArrayOutputStream content = new ByteArrayOutputStream();
            try {
                entity.writeTo(content);
            } catch (IOException e) {
                throw new RuntimeException("Failed to copy request body", e);
            }
            // Replace the now-consumed body with a copy of the content stream
            byte[] body = content.toByteArray();
            if (consumed) {
                request.setEntity(new ByteArrayEntity(body));
            }
            return body;
        }
    }
}
```

NODEJS

```
/*
    Version 1.0.1
    Before running this example, install necessary dependencies by running:
    npm install http-signature jssha
*/

var fs = require('fs');
```

```
var https = require('https');
var os = require('os');
var httpSignature = require('http-signature');
var jsSHA = require("jssha");


// TODO: update these values to your own
var tenancyId = "ocid1.tenancy.oc1..aaaaaaaaba3pv6wkcr4jqae5f15p2b2m2yt2j6rx32uzr4h25vqstifsfdsq";
var authUserId = "ocid1.user.oc1..aaaaaaaat5nvwcna5j6aqzjcaty5eqbb6qt2jvpkanghtgdaqedqw3rynjq";
var keyFingerprint = "20:3b:97:13:55:1c:5b:0d:d3:37:d8:50:4e:c5:3a:34";
var privateKeyPath = "~/.oci/oci_api_key.pem";


var identityDomain = "identity.us-ashburn-1.oraclecloud.com";
var coreServicesDomain = "iaas.us-ashburn-1.oraclecloud.com";


if(privateKeyPath.indexOf("~/") === 0) {
    privateKeyPath = privateKeyPath.replace("~", os.homedir())
}
var privateKey = fs.readFileSync(privateKeyPath, 'ascii');


// signing function as described at
https://docs.cloud.oracle.com/Content/API/Concepts/signingrequests.htm
                              function sign(request, options) {

    var apiKeyId = options.tenancyId + "/" + options.userId + "/" + options.keyFingerprint;

    var headersToSign = [
        "host",
        "date",
        "(request-target)"
    ];

    var methodsThatRequireExtraHeaders = ["POST", "PUT"];

    if(methodsThatRequireExtraHeaders.indexOf(request.method.toUpperCase()) !== -1) {
        options.body = options.body || "";

        var shaObj = new jsSHA("SHA-256", "TEXT");
        shaObj.update(options.body);
```

```
        request.setHeader("Content-Length", options.body.length);
        request.setHeader("x-content-sha256", shaObj.getHash('B64'));

        headersToSign = headersToSign.concat([
            "content-type",
            "content-length",
            "x-content-sha256"
        ]);
    }

    httpSignature.sign(request, {
        key: options.privateKey,
        keyId: apiKeyId,
        headers: headersToSign
    });

    var newAuthHeaderValue = request.getHeader("Authorization").replace("Signature ", "Signature
version=\"1\",");
    request.setHeader("Authorization", newAuthHeaderValue);
}

// generates a function to handle the https.request response object
function handleRequest(callback) {

    return function(response) {
        var responseBody = "";

        response.on('data', function(chunk) {
        responseBody += chunk;
    });

        response.on('end', function() {
            callback(JSON.parse(responseBody));
        });
    }
}

// gets the user with the specified id
function getUser(userId, callback) {

    var options = {
```

```
        host: identityDomain,
        path: "/20160918/users/" + encodeURIComponent(userId),
    };

    var request = https.request(options, handleRequest(callback));

    sign(request, {
        privateKey: privateKey,
        keyFingerprint: keyFingerprint,
        tenancyId: tenancyId,
        userId: authUserId
    });

    request.end();
};

// creates a Oracle Cloud Infrastructure VCN in the specified compartment
function createVCN(compartmentId, displayName, cidrBlock, callback) {

    var body = JSON.stringify({
        compartmentId: compartmentId,
        displayName: displayName,
        cidrBlock: cidrBlock
    });

    var options = {
        host: coreServicesDomain,
        path: '/20160918/vcns',
        method: 'POST',
        headers: {
            "Content-Type": "application/json",
        }
    };

    var request = https.request(options, handleRequest(callback));

    sign(request, {
        body: body,
        privateKey: privateKey,
        keyFingerprint: keyFingerprint,
        tenancyId: tenancyId,
        userId: authUserId
```

```
    });

    request.end(body);
};

// test the above functions
console.log("GET USER:");

getUser(authUserId, function(data) {
    console.log(data);

    console.log("\nCREATING VCN:");

    // TODO: replace this with a compartment you have access to
    var compartmentIdToCreateVcnIn = tenancyId;

    createVCN(compartmentIdToCreateVcnIn, "Test-VCN", "10.0.0.0/16", function(data) {
        console.log(data);
    });
});
```

PERL

This sample omits the optional `version` field in the Authorization header.

```
#!/usr/bin/perl
# Version 1.0.1
# Need the following:
# Modules - Authen::HTTP::Signature, DateTime, DateTime::Format::HTTP, Mozilla::CA, File::Slurp,
LWP::UserAgent, LWP::Protocol::https
# LWP::UserAgent and LWP::Protoco::https >= 6.06
# OpenSSL >= 1.0.1

use strict;
use warnings;

{
   package OCISigner;

   use Authen::HTTP::Signature;
   use Digest::SHA qw(sha256_base64);
   use DateTime;
   use DateTime::Format::HTTP;
```

```
my @generic_headers = (
  'date', '(request-target)', 'host'
);
my @body_headers = (
  'content-length', 'content-type', 'x-content-sha256'
);
my @all_headers = (@generic_headers, @body_headers);
my %required_headers = (
  get => \@generic_headers,
  delete => \@generic_headers,
  head => \@generic_headers,
  post => \@all_headers,
  put => \@all_headers
);

sub new {
   my ( $class, $api_key, $private_key) = @_;
   my $self = {
       _api_key => $api_key,
       _private_key  => $private_key
   };
   bless $self, $class;
   return $self;
}

sub sign_request {
   my ( $self, $request ) = @_;
   my $verb = lc($request->method);
   my $sign_body = grep(/^$verb$/, ('post', 'put'));
   $self->inject_missing_headers($request, $sign_body);
   my $headers = $required_headers{$verb};

   my $all_auth = Authen::HTTP::Signature->new(
     key => $self->{_private_key},
     request => $request,
     key_id => $self->{_api_key},
     headers => $headers,
   );
   $all_auth->sign();
}
```

```
   sub inject_missing_headers {
      my ( $self, $request, $sign_body ) = @_;
      $request->header('content-type', 'application/json') unless $request->header('content-type');
      $request->header('accept', '*/*') unless $request->header('accept');
      my $class = 'DateTime::Format::HTTP';
      $request->header('date', $class->format_datetime(DateTime->now)) unless $request->header('date');

      $request->header('host', $request->uri->host) unless $request->header('host');
      if ($sign_body) {
        $request->content('') unless $request->content;
        $request->header('content-length', length($request->content)) unless $request->header('content-
length');
        $request->header('x-content-sha256', $self->compute_sha256($request->content)) unless $request-
>header('x-content-sha256');
      }
   }

   sub compute_sha256 {
      my ( $self, $content ) = @_;
      my $digest = sha256_base64($content);
      while (length($digest) % 4) {
         $digest .= '=';
      }
      return $digest;
   }
} # OCISigner

{
   package OCIClient;

   use LWP::UserAgent;
   use Mozilla::CA;

   sub new {
      my ( $class, $api_key, $private_key ) = @_;
      my $ua = LWP::UserAgent->new;
      $ua->ssl_opts(
        verify_hostname => 1,
        SSL_ca_file => Mozilla::CA::SSL_ca_file()
      );
      my $self = {
        _signer => OCISigner->new($api_key, $private_key),
```

```
        _ua => $ua
      };
      bless $self, $class;
      return $self;
   }

   sub make_request {
     my ( $self, $request ) = @_;
     print "Sending request\n";
     $self->{_signer}->sign_request($request);

     my $response = $self->{_ua}->request($request);
     if ($response->is_success) {
       my $message = $response->decoded_content;
       print "Received reply: $message\n";
     }
     else {
       print "HTTP GET error code: ", $response->code, "\n";
       print "HTTP GET error message: ", $response->message, "\n";
     }
   }
} # OCIClient

use File::Slurp qw(read_file);

my $api_key = "ocid1.tenancy.oc1..aaaaaaaaba3pv6wkcr4jqae5f15p2b2m2yt2j6rx32uzr4h25vqstifsfdsq/" .
              "ocid1.user.oc1..aaaaaaaat5nvwcna5j6aqzjcaty5eqbb6qt2jvpkanghtgdaqedqw3rynjq/" .
              "20:3b:97:13:55:1c:5b:0d:d3:37:d8:50:4e:c5:3a:34";

my $private_key = read_file('../sample-private-key') or die $!;

my $client = OCIClient->new($api_key, $private_key);

# Uncomment to use a fixed date
#my $fixed_date = 'Thu, 05 Jan 2014 21:31:40 GMT';
my $fixed_date;

# GET with query parameters
# Note: Older ocid formats included ":" which must be escaped
my %query_args = (
   availability_domain => "Pjwf%3A%20PHX-AD-1",
   compartment_id =>
```

```
"ocid1.compartment.oc1..aaaaaaaam3we6vgnherjq5q2idnccdflvjsnog7mlr6rtdb25gilchfeyjxa",
   display_name => "TeamXInstances",
   volume_id => "ocid1.volume.oc1.phx.abyhqljrgvttnlx73nmrwfaux7kcvzfs3s66izvxf2h4lgvyndsdsnoiwr5q"
);
my $uri = "https://iaas.us-phoenix-1.oraclecloud.com/20160918/instances?availabilityDomain=" .
          $query_args{availability_domain} .
          "&compartmentId=" .
          $query_args{compartment_id} .
          "&displayName=" .
          $query_args{display_name} .
          "&volumeId=" .
          $query_args{volume_id};
my $request = HTTP::Request->new(GET => $uri);
$request->header('date', $fixed_date) if $fixed_date;
$client->make_request($request);

# POST with body
$uri = "https://iaas.us-ashburn-1.oraclecloud.com/20160918/volumeAttachments";
my $body = q|{
    "compartmentId":
"ocid1.compartment.oc1..aaaaaaaam3we6vgnherjq5q2idnccdflvjsnog7mlr6rtdb25gilchfeyjxa",
    "instanceId": "ocid1.instance.oc1.phx.abuw4ljrlsfiqw6vzzxb43vyypt4pkodawglp3wqxjqofakrwvou52gb6s5a",
    "volumeId": "ocid1.volume.oc1.phx.abyhqljrgvttnlx73nmrwfaux7kcvzfs3s66izvxf2h4lgvyndsdsnoiwr5q"
}|;
$request = HTTP::Request->new(POST => $uri);
$request->header('date', $fixed_date) if $fixed_date;
$request->content($body);
$client->make_request($request);
```

PHP

.

```
<? php

// Version 1.0.0
//
//  Dependencies:
//  - PHP curl extension
//  - Guzzle (composer require guzzlehttp/guzzle)
//

require 'vendor/autoload.php';
```

```
use Psr\Http\Message\RequestInterface;
use GuzzleHttp\HandlerStack;
use GuzzleHttp\Handler\CurlHandler;
use GuzzleHttp\Client;
use GuzzleHttp\Middleware;

// TODO: Update these for your tenancy
$tenancy_id = 'ocid1.tenancy.oc1..aaaaaaaaba3pv6wkcr4jqae5f15p2b2m2yt2j6rx32uzr4h25vqstifsfdsq';
$user_id = 'ocid1.user.oc1..aaaaaaaat5nvwcna5j6aqzjcaty5eqbb6qt2jvpkanghtgdaqedqw3rynjq';
$thumbprint = '20:3b:97:13:55:1c:5b:0d:d3:37:d8:50:4e:c5:3a:34';
$region = 'us-phoenix-1';
$key_location = 'file://private.pem';
$key_passphrase = 'password';

$namespace = 'MyNamespace';
$bucket_name = 'MyBucket';
$file_to_upload = 'myfile.txt';

$key_id = "$tenancy_id/$user_id/$thumbprint";

function sign_string($data, $key_path, $passphrase){
    $pkeyid = openssl_pkey_get_private($key_path, $passphrase);
    if (!$pkeyid) {
       exit('Error reading private key');
       }

       openssl_sign($data, $signature, $pkeyid, OPENSSL_ALGO_SHA256);

       // free the key from memory
       openssl_free_key($pkeyid);

       return base64_encode($signature);
}

function oci_signer_middleware(RequestInterface $request) {
       global $key_id;
       global $key_location;
       global $key_passphrase;

       // headers required for all HTTP verbs
       $headers = "date (request-target) host";
```

```
      // example: Thu, 05 Jan 2014 21:31:40 GMT
      $date=gmdate("D, d M Y H:i:s T", time());
      $method = strtolower($request->getMethod());
      $request_target = $request->getRequestTarget();
      $host = $request->getHeader('Host')[0];

      $request = $request->withHeader('Date', $date);

      $signing_string = "date: $date\n(request-target): $method $request_target\nhost: $host";

      // additional required headers for POST and PUT requests
      if ($method == 'post' || $method == 'put') {
         $content_length = $request->getHeader('Content-Length')[0];

         // if content length is 0 we still need to explicitly send the Content-Length header
         if (!$content_length){
             $content_length = 0;
                 $request = $request->withHeader('Content-Length', 0);
      }

         $content_type = $request->getHeader('Content-Type')[0];
         $content_sha256 = base64_encode(hex2bin(hash("sha256", $request->getBody())));

         $request = $request->withHeader('x-content-sha256', $content_sha256);

         $headers = $headers . " content-length content-type x-content-sha256";
         $signing_string = $signing_string . "\ncontent-length: $content_length\ncontent-type: $content_
type\nx-content-sha256: $content_sha256";
      }

      echo "Signing string:\n$signing_string".PHP_EOL;

      $signature = sign_string($signing_string, $key_location, $key_passphrase);

      $authorization_header = "Signature version=\"1\",keyId=\"$key_id\",algorithm=\"rsa-
sha256\",headers=\"$headers\",signature=\"$signature\"";
      $request = $request->withHeader('Authorization', $authorization_header);

      echo "\nRequest headers:".PHP_EOL;
      foreach ($request->getHeaders() as $name => $values) {
         echo $name . ': ' . implode(', ', $values) . "\n";
```

```
        }

        return $request;
}

// EXAMPLE REQUESTS
$handler = new CurlHandler();
$stack = HandlerStack::create($handler);

// place signing middleware after prepare-body so it can access Content-Length header
$stack->after('prepare_body', Middleware::mapRequest('oci_signer_middleware'));

$client = new Client([
    'handler' => $stack
]);

// GET current user
echo "************************************".PHP_EOL;
echo "Getting user: $user_id...".PHP_EOL;
echo "************************************".PHP_EOL;
$response = $client->get("https://identity.$region.oraclecloud.com/20160918/users/$user_id");
echo "\nResponse:\n";
echo $response->getStatusCode().PHP_EOL;
echo $response->getBody().PHP_EOL.PHP_EOL;

// Create a VCN
echo "************************************".PHP_EOL;
echo "Creating VCN...".PHP_EOL;
echo "************************************".PHP_EOL;
$body = "{\"cidrBlock\" : \"10.0.0.0/16\",\"compartmentId\" : \"$tenancy_id\",\"displayName\" :
\"MyPhpVcn\"}";
$response = $client->post("https://iaas.$region.oraclecloud.com/20160918/vcns", [ "body" => $body,
'headers' => ['Content-Type' => 'application/json']]);
echo "\nResponse:".PHP_EOL;
echo $response->getStatusCode().PHP_EOL;
echo $response->getBody().PHP_EOL.PHP_EOL;

// PUT object with no content
echo "************************************".PHP_EOL;
echo "Putting object 'NewObject'...".PHP_EOL;
echo "************************************".PHP_EOL;
$body = '';
```

```
$response = $client->put("https://objectstorage.$region.oraclecloud.com/n/$namespace/b/$bucket_
name/o/NewObject", [ "body" => $body, 'headers' => ['Content-Type' => 'application/json']]);
echo "\nResponse:\n";
echo $response->getStatusCode().PHP_EOL;
echo $response->getBody().PHP_EOL;

// PUT object with content
echo "************************************".PHP_EOL;
echo "Putting object 'NewObject2'...".PHP_EOL;
echo "************************************".PHP_EOL;

$file_handle = fopen($file_to_upload, "rb");
$body = "";
while (!feof($file_handle)) {
   $body = $body . fgets($file_handle);
}
fclose($file_handle);

$response = $client->put("https://objectstorage.$region.oraclecloud.com/n/$namespace/b/$bucket_
name/o/NewObject2", [ "body" => $body, 'headers' => ['Content-Type' => 'application/octet-stream']]);
echo "\nResponse:\n";
echo $response->getStatusCode().PHP_EOL;
echo $response->getBody().PHP_EOL;

?>
```

PYTHON

This sample omits the optional `version` field in the Authorization header.

> **Important**
>
> This Python sample code requires TLS 1.2, which is not included with the default Python on Mac OS X.

```
import base64
import email.utils
import hashlib

# pip install httpsig_cffi requests six
```

```
import httpsig_cffi.sign
import requests
import six
# Version 1.0.1

class SignedRequestAuth(requests.auth.AuthBase):
    """A requests auth instance that can be reused across requests"""
    generic_headers = [
        "date",
        "(request-target)",
        "host"
    ]
    body_headers = [
        "content-length",
        "content-type",
        "x-content-sha256",
    ]
    required_headers = {
        "get": generic_headers,
        "head": generic_headers,
        "delete": generic_headers,
        "put": generic_headers + body_headers,
        "post": generic_headers + body_headers
    }

    def __init__(self, key_id, private_key):
        # Build a httpsig_cffi.requests_auth.HTTPSignatureAuth for each
        # HTTP method's required headers
        self.signers = {}
        for method, headers in six.iteritems(self.required_headers):
            signer = httpsig_cffi.sign.HeaderSigner(
                key_id=key_id, secret=private_key,
                algorithm="rsa-sha256", headers=headers[:])
            use_host = "host" in headers
            self.signers[method] = (signer, use_host)

    def inject_missing_headers(self, request, sign_body):
        # Inject date, content-type, and host if missing
        request.headers.setdefault(
            "date", email.utils.formatdate(usegmt=True))
        request.headers.setdefault("content-type", "application/json")
        request.headers.setdefault(
```

```
            "host", six.moves.urllib.parse.urlparse(request.url).netloc)

        # Requests with a body need to send content-type,
        # content-length, and x-content-sha256
        if sign_body:
            body = request.body or ""
            if "x-content-sha256" not in request.headers:
                m = hashlib.sha256(body.encode("utf-8"))
                base64digest = base64.b64encode(m.digest())
                base64string = base64digest.decode("utf-8")
                request.headers["x-content-sha256"] = base64string
            request.headers.setdefault("content-length", len(body))

    def __call__(self, request):
        verb = request.method.lower()
        # nothing to sign for options
        if verb == "options":
            return request
        signer, use_host = self.signers.get(verb, (None, None))
        if signer is None:
            raise ValueError(
                "Don't know how to sign request verb {}".format(verb))

        # Inject body headers for put/post requests, date for all requests
        sign_body = verb in ["put", "post"]
        self.inject_missing_headers(request, sign_body=sign_body)

        if use_host:
            host = six.moves.urllib.parse.urlparse(request.url).netloc
        else:
            host = None

        signed_headers = signer.sign(
            request.headers, host=host,
            method=request.method, path=request.path_url)
        request.headers.update(signed_headers)
        return request


# -----BEGIN RSA PRIVATE KEY-----
# ...
# -----END RSA PRIVATE KEY-----
```

```
with open("../sample-private-key") as f:
    private_key = f.read().strip()

# This is the keyId for a key uploaded through the console
api_key = "/".join([
    "ocid1.tenancy.oc1..aaaaaaaaba3pv6wkcr4jqae5f15p2b2m2yt2j6rx32uzr4h25vqstifsfdsq",
    "ocid1.user.oc1..aaaaaaaat5nvwcna5j6aqzjcaty5eqbb6qt2jvpkanghtgdaqedqw3rynjq",
    "20:3b:97:13:55:1c:5b:0d:d3:37:d8:50:4e:c5:3a:34"
])

auth = SignedRequestAuth(api_key, private_key)

headers = {
    "content-type": "application/json",
    "date": email.utils.formatdate(usegmt=True),
    # Uncomment to use a fixed date
    # "date": "Thu, 05 Jan 2014 21:31:40 GMT"
}


# GET with query parameters
uri = "https://iaas.us-ashburn-1.oraclecloud.com/20160918/instances?availabilityDomain={availability_
domain}&compartmentId={compartment_id}&displayName={display_name}&volumeId={volume_id}"
uri = uri.format(
    availability_domain="Pjwf%3A%20PHX-AD-1",
    # Older ocid formats included ":" which must be escaped
    compartment_
id="ocid1.compartment.oc1..aaaaaaaam3we6vgnherjq5q2idnccdflvjsnog7mlr6rtdb25gilchfeyjxa".replace(":",
"%3A"),
    display_name="TeamXInstances",
    volume_
id="ocid1.volume.oc1.phx.abyhqljrgvttnlx73nmrwfaux7kcvzfs3s66izvxf2h4lgvyndsdsnoiwr5q".replace(":",
"%3A")
)
response = requests.get(uri, auth=auth, headers=headers)
print(uri)
print(response.request.headers["Authorization"])


# POST with body
uri = "https://iaas.us-ashburn-1.oraclecloud.com/20160918/volumeAttachments"
body = """{
```

```
    "compartmentId":
"ocid1.compartment.oc1..aaaaaaaam3we6vgnherjq5q2idnccdflvjsnog7mlr6rtdb25gilchfeyjxa",
    "instanceId": "ocid1.instance.oc1.phx.abuw4ljrlsfiqw6vzzxb43vyypt4pkodawglp3wqxjqofakrwvou52gb6s5a",
    "volumeId": "ocid1.volume.oc1.phx.abyhqljrgvttnlx73nmrwfaux7kcvzfs3s66izvxf2h4lgvyndsdsnoiwr5q"
}"""
response = requests.post(uri, auth=auth, headers=headers, data=body)
print("\n" + uri)
print(response.request.headers["Authorization"])
```

RUBY

```
require 'base64'
require 'digest'
require 'openssl'
require 'time'
require 'uri'

# gem 'httparty', '~> 0.13.0'
require 'httparty'

# Version 1.0.1
class Client
    include HTTParty
    attr_reader :signer

    def initialize(key_id, private_key)
        @signer = Signer.new(key_id, private_key)
    end

    # nothing to sign for :options

    [:get, :head, :delete].each do |method|
        define_method(method) do |uri, headers: {}|
            self.signer.sign(method, uri, headers, body: nil)
            self.class.send(method, uri, :headers => headers)
        end
    end

    [:put, :post].each do |method|
        define_method(method) do |uri, headers: {}, body: ""|
            self.signer.sign(method, uri, headers, body)
            self.class.send(method, uri, :headers => headers, :body => body)
        end
```

```
    end
end


class Signer
    class << self
        attr_reader :headers
    end

    attr_reader :key_id, :private_key

    generic_headers = [:"date", :"(request-target)", :"host"]
    body_headers = [
        :"content-length", :"content-type", :"x-content-sha256"]
    @headers = {
        get: generic_headers,
        head: generic_headers,
        delete: generic_headers,
        put: generic_headers + body_headers,
        post: generic_headers + body_headers
    }

    def initialize(key_id, private_key)
        @key_id = key_id
        @private_key = private_key
    end

    def sign(method, uri, headers, body)
        uri = URI(uri)
        path = uri.query.nil? ? uri.path : "#{uri.path}?#{uri.query}"
        self.inject_missing_headers(headers, method, body, uri)
        signature = self.compute_signature(headers, method, path)
        unless signature.nil?
            self.inject_authorization_header(headers, method, signature)
        end
    end

    def inject_missing_headers(headers, method, body, uri)
        headers["content-type"] ||= "application/json"
        headers["date"] ||= Time.now.utc.httpdate
        headers["accept"] ||= "*/*"
        headers["host"] ||= uri.host
```

```
        if method == :put or method == :post
            body ||= ""
            headers["content-length"] ||= body.length.to_s
            headers["x-content-sha256"] ||= Digest::SHA256.base64digest(body)
        end
    end

    def inject_authorization_header(headers, method, signature)
        signed_headers = self.class.headers[method].map(&:to_s).join(" ")
        headers["authorization"] = [
            %(Signature version="1"),
            %(headers="#{signed_headers}"),
            %(keyId="#{self.key_id}"),
            %(algorithm="rsa-sha256"),
            %(signature="#{signature}")
        ].join(",")
    end

    def compute_signature(headers, method, path)
        return if self.class.headers[method].empty?
        signing_string = self.class.headers[method].map do |header|
            if header == :"(request-target)"
                "#{header}: #{method.downcase} #{path}"
            else
                "#{header}: #{headers[header.to_s]}"
            end
        end.join("\n")
        signature = self.private_key.sign(
            OpenSSL::Digest::SHA256.new,
            signing_string.encode("ascii"))
        Base64.strict_encode64(signature)
    end
end


api_key = [
    "ocid1.tenancy.oc1..aaaaaaaaba3pv6wkcr4jqae5f15p2b2m2yt2j6rx32uzr4h25vqstifsfdsq",
    "ocid1.user.oc1..aaaaaaaat5nvwcna5j6aqzjcaty5eqbb6qt2jvpkanghtgdaqedqw3rynjq",
    "20:3b:97:13:55:1c:5b:0d:d3:37:d8:50:4e:c5:3a:34"
].join("/")
private_key = OpenSSL::PKey::RSA.new(File.read("../sample-private-key"))
client = Client.new(api_key, private_key)
```

```
headers = {
    # Uncomment to use a fixed date
    # "date" => "Thu, 05 Jan 2014 21:31:40 GMT"
}

# GET with query parameters
uri = "https://iaas.us-ashburn-1.oraclecloud.com/20160918/instances?availabilityDomain=%{availability_
domain}&compartmentId=%{compartment_id}&displayName=%{display_name}&volumeId=%{volume_id}"
uri = uri % {
    :availability_domain => "Pjwf%3A%20PHX-AD-1",
    # Older ocid formats included ":" which must be escaped
    :compartment_id =>
"ocid1.compartment.oc1..aaaaaaaam3we6vgnherjq5q2idnccdflvjsnog7mlr6rtdb25gilchfeyjxa".sub(":", "%3A"),
    :display_name => "TeamXInstances",
    :volume_id =>
"ocid1.volume.oc1.phx.abyhqljrgvttnlx73nmrwfaux7kcvzfs3s66izvxf2h4lgvyndsdsnoiwr5q".sub(":", "%3A")
}
response = client.get(uri, headers: headers)
puts uri
puts response.request.options[:headers]["authorization"]
puts response.response

# POST with body
uri = "https://iaas.us-ashburn.oraclecloud.com/20160918/volumeAttachments"
body = %q({
    "compartmentId":
"ocid1.compartment.oc1..aaaaaaaam3we6vgnherjq5q2idnccdflvjsnog7mlr6rtdb25gilchfeyjxa",
    "instanceId": "ocid1.instance.oc1.phx.abuw4ljrlsfiqw6vzzxb43vyypt4pkodawglp3wqxjqofakrwvou52gb6s5a",
    "volumeId": "ocid1.volume.oc1.phx.abyhqljrgvttnlx73nmrwfaux7kcvzfs3s66izvxf2h4lgvyndsdsnoiwr5q"
})
response = client.post(uri, headers: headers, body: body)
puts "\n" + uri
puts response.request.options[:headers]["authorization"]
puts response.response
```

Go

The following example shows how to create a default signer.



Note

The Go SDK exposes a stand-alone signer that you can use to sign custom requests. You can find related code at http_signer.go.

```
client := http.Client{}
var request http.Request
request = ... // some custom request

// Set the Date header
request.Header.Set("Date", time.Now().UTC().Format(http.TimeFormat))

// And a provider of cryptographic keys
provider := common.DefaultConfigProvider()

// Build the signer
signer := common.DefaultSigner(provider)

// Sign the request
signer.Sign(&request)

// Execute the request
client.Do(request)
```

The following example shows how the signer can allow more granular control on the headers used for signing:

```
client := http.Client{}
var request http.Request
request = ... // some custom request

// Set the Date header
request.Header.Set("Date", time.Now().UTC().Format(http.TimeFormat))

// Mandatory headers to be used in the sign process
defaultGenericHeaders    = []string{"date", "(request-target)", "host"}

// Optional headers
optionalHeaders = []string{"content-length", "content-type", "x-content-sha256"}
```

```
// A predicate that specifies when to use the optional signing headers
optionalHeadersPredicate := func (r *http.Request) bool {
       return r.Method == http.MethodPost
}

// And a provider of cryptographic keys
provider := common.DefaultConfigProvider()

// Build the signer
signer := common.RequestSigner(provider, defaultGenericHeaders, optionalHeaders,
optionalHeadersPredicate)

// Sign the request
signer.Sign(&request)

// Execute the request
c.Do(request)
```

GLOSSARY

A

AD-specific subnet

A subnet that is specific to a particular availability domain (AD). Historically all subnets were AD-specific. Compare with regional subnets, which Oracle recommends over AD-specific subnets.

alarm

The trigger rule and query to evaluate and related configuration, such as notification details to use when the trigger is breached. Alarms passively monitor your cloud resources using metrics in Monitoring.

API key

A credential for securing requests to the Oracle Cloud Infrastructure REST API.

attach

Link a volume and instance together. Allows an instance to connect and mount the volume as a hard drive.

auth token

Oracle Cloud Infrastructure-generated token you use to authenticate with third-party APIs, such as a Swift client.

availability domain

One or more isolated, fault-tolerant Oracle data centers that host cloud resources such as instances, volumes, and subnets. A region contains one or more availability domains.

B

backend set

A logical entity defined by a list of backend servers, a load balancing policy, and a health check policy.

bare metal IaaS

A cloud infrastructure that allows you to utilize hosted physical hardware, as opposed to traditional software-based virtual machines, ensuring a high level of security and performance.

block storage volume

A virtual disk that provides persistent storage space for instances in the cloud.

bucket

A logical container for storing objects.

C

CHAP

Stands for Challenge-Handshake-Authentication-Protocol. It is a security protocol used by iSCSI for authentication between a volume and an instance.

Cloud Block Storage

A service that allows you to add block storage volumes to an instance in order to expand the available storage on that resource.

cloud network

A virtual version of a traditional network—including CIDRs, subnets, route tables, and gateways—on which your instance runs.

compartment

A collection of related resources that can be accessed only by certain groups that have been given permission by an administrator in your organization.

Compute

A service that lets you provision and manage compute hosts, known as instances.

connect

Make an attached volume usable by an instance's guest OS.

CPE

A virtual representation of the edge router at your end of an IPSec VPN that connects your VCN and on-premises network.

cross-connect

Used with Oracle Cloud Infrastructure FastConnect. In a colocation scenario, this is the physical cable connecting your existing network to Oracle in the FastConnect location.

cross-connect group

Used with Oracle Cloud Infrastructure FastConnect. In a colocation scenario, this is a link aggregation group (LAG) that contains at least one cross-connect.

customer-premises equipment

A virtual representation of the edge router at your end of an IPSec VPN that connects your VCN and on-premises network.

D

data point

A timestamp-value pair for the specified metric. (Monitoring service.) Example: 2018-05-10T22:19:00Z, 10.4

DB System

A dedicated bare metal instance running Oracle Linux, optimized for running one or more Oracle databases. A DB System is a Database Service resource.

DHCP options

Configuration information that is automatically provided to the instances when they boot up.

dimension

A qualifier provided in a metric definition. (Monitoring service.) Example: Resource identifier (resourceId), provided in the definitions of oci_computeagent metrics.

display name

A friendly name or description that helps you easily identify the resource.

DRG

An optional virtual router that you can add to your VCN to provide a path for private network traffic between your VCN and on-premises network.

DRG attachment

When you attach a dynamic routing gateway (DRG) to a virtual cloud network (VCN), the result is a DRG attachment object. To detach the DRG, you delete that attachment object.

dynamic group

A special type of IAM group that contains instances that match rules that you define (thus the membership can change dynamically as matching instances are terminated or launched). These instances act as "principal" actors and can make API calls to Oracle Cloud Infrastructure services according to IAM policies that you write for the dynamic group.

dynamic routing gateway

An optional virtual router that you can add to your VCN to provide a path for private network traffic between your VCN and on-premises network.

E

ephemeral public IP

A public IP address (and related properties) that is temporary and exists for the life of the instance it's assigned to. It can be assigned only to the primary private IP on a VNIC. Compare with reserved public IP.

Export

Controls how file systems are accessed by NFS clients when they connect to a mount target.

Export Options

A set of parameters that specify the level of access granted to NFS clients when they connect to a mount target.

F

FastConnect

FastConnect provides an easy way to create a dedicated, private connection between your data center or existing network and Oracle Cloud Infrastructure. FastConnect provides higher-bandwidth options, and a more reliable and consistent networking experience compared to internet-based connections.

fault domain

A logical grouping of hardware and infrastructure within an availability domain to provide isolation of resources in case of hardware failure or unexpected software changes.

File System

An organized system of directories and folders where data is stored.

frequency

The time period between each posted raw data point for a given metric. (Raw data points are posted by the metric namespace to the Monitoring service.)

G

group

A collection of users who all need a particular type of access to a set of resources or compartment.

guest operating system

An operating system installed on a cloud instance.

guest OS

An operating system installed on a cloud instance.

H

health check

A test to confirm the availability of backend servers.

I

IaaS

A service that allows customers to rapidly scale up or down their computer infrastructure (computing, storage, or network).

IAM

The service for controlling authentication and authorization of users who need to use your cloud resources. Also called "IAM".

Identity and Access Management Service

The service for controlling authentication and authorization of users who need to use your cloud resources. Also called "IAM".

identity provider

A service that provides identifying credentials and authentication for federated users.

IdP

Short for "identity provider", which is a service that provides identifying credentials and authentication for federated users.

image

A template of a virtual hard drive that determines the operating system and other software for an instance.

Infrastructure-as-a-Service

A service that allows customers to rapidly scale up or down their computer infrastructure (computing, storage, or network).

instance

A bare metal or virtual machine (VM) compute host. The image used to launch the instance determines its operating system and other software. The shape specified during the launch process determines the number of CPUs and memory allocated to the instance.

internet gateway

An optional virtual router that you can add to your VCN. It provides a path for network traffic between your VCN and the internet.

interval

The time window used to convert the given set of raw data points. (Monitoring service.) Example: 5 minutes

IPSec connection

The secure connection between a dynamic routing gateway (DRG) and customer-premises equipment (CPE), consisting of multiple IPSec tunnels. The IPSec connection is one of the components forming a site-to-site VPN between a virtual cloud network (VCN) and your on-premises network.

IQN

A unique ID assigned to an iSCSI device. Used when connecting a volume to an instance.

iSCSI

A TCP/IP based standard used for communication between a volume and attached instance.

iSCSI Qualified Name

A unique ID assigned to an iSCSI device. Used when connecting a volume to an instance.

K

key pair

A security mechanism consisting of a public key and a private key. Required (for example) for Secure Shell (SSH) access to an instance.

L

listener

An entity that checks for incoming traffic on the load balancer's public floating IP address.

local peering gateway

A component on a VCN for routing traffic to a locally peered VCN. "Local" peering means the two VCNs are in the same region. Compare with a remote peering connection.

local VCN peering

The process of connecting two VCNs in the same region so that their resources can communicate without routing the traffic over the internet or through your on-premises network.

LPG

A component on a VCN for routing traffic to a locally peered VCN. "Local" peering means the two VCNs are in the same region. Compare with a remote peering connection.

M

message

An alert published to all subscriptions in the specified topic. Each message is delivered at least once per subscription. (Notifications and Monitoring services.)

metric

A measurement related to health, capacity, or performance of a given resource. (Monitoring service). Example: CpuUtilization

metric definition

A set of references, qualifiers, and other information provided by a metric namespace for a given metric. (Monitoring service.)

metric namespace

Indicator of the resource, service, or application that emits the metric. Provided in the metric definition. (Monitoring service.) Example: oci_computeagent

metric stream

An individual set of aggregated data for a metric. Typically specific to a resource. (Monitoring service.)

Monitoring Query Language

The syntax used for metric and alarm queries. (Monitoring service.)

Mount Point

A directory from which a client may access a remote File Storage Service file system.

Mount Target

An NFS endpoint that allows a file system to be accessed by clients.

MQL

Monitoring Query Language.The syntax used for metric and alarm queries. In the Console, MQL syntax of queries is displayed in Advanced Mode. (Monitoring service.)

N

NAT gateway

An optional virtual router that you can add to your VCN to perform Network Address Translation (NAT). A NAT gateway gives cloud resources without public IP addresses access to the internet without exposing those resources to incoming internet connections.

notification destination

Protocol and other details for sending messages when the alarm transitions to another state, such as from "OK" to "FIRING." (Monitoring service.)

O

object

Any type of data, regardless of content type, is stored as an object. The object is composed of the object itself and metadata about the object. Each object is stored in a bucket.

OCID

An Oracle-assigned unique ID called an Oracle Cloud Identifier (OCID). This ID is included as part of the resource's information in both the Console and API.

one-time password

A single-use Console password that Oracle assigns to a new user, or to an existing user who requested a password reset.

Oracle Cloud Identifier

An Oracle-assigned unique ID called an Oracle Cloud Identifier (OCID). This ID is included as part of the resource's information in both the Console and API.

OTP

A single-use Console password that Oracle assigns to a new user, or to an existing user who requested a password reset.

P

policy

An IAM document that specifies who has what type of access to your resources. It is used in different ways: to mean an individual statement written in the policy language; to mean a collection of statements in a single, named "policy" document (which has an Oracle Cloud ID (OCID) assigned to it); and to mean the overall body of policies your organization uses to control access to resources.

policy statement

Policies can contain one or more individual statements. Each statement gives a group a certain type of access to certain resources in a particular compartment.

primary IP

The private IP that is automatically created and assigned to a VNIC during creation.

primary VNIC

The VNIC that is automatically created and attached to an instance during launch.

private IP

An object that contains a private IP address and related properties such as a hostname for DNS. Each instance automatically comes with a primary private IP, and you can add secondary ones.

private peering

One of the ways to use FastConnect. Private peering lets you extend your existing infrastructure into a virtual cloud network (VCN) in Oracle Cloud Infrastructure (for example, to implement a hybrid cloud, or a lift and shift scenario). Communication across the connection is with IPv4 private addresses (typically RFC 1918).

private subnet

A subnet in which instances are not allowed to have public IP addresses

private virtual circuit

A FastConnect virtual circuit that supports private peering.

public IP

An object that contains a public IP address and related properties. You control whether each private IP on an instance has an assigned public IP. There are two types: reserved public IPs and ephemeral public IPs.

public peering

One of the way to use FastConnect. Public peering lets your on-premises network access public services in Oracle Cloud Infrastructure without using the internet. For example, Object Storage, the Oracle Cloud Infrastructure Console and APIs, or public load balancers in your VCN. Communication across the connection is with IPv4 public IP addresses. Without FastConnect, the traffic destined for public IP addresses would be routed over the internet. With FastConnect, that traffic goes over your private physical connection.

public subnet

A subnet in which instances are allowed to have public IP addresses. When you launch an instance in a public subnet, you specify whether the instance should have a public IP address.

public virtual circuit

A FastConnect virtual circuit that supports public peering.

Q

query

The expression to evaluate for returning aggregated data. A valid query includes a metric, statistic, and interval. In the Console, you can view a query in Basic Mode or Advanced Mode. The latter displays the Monitoring Query Language (MQL) syntax. (Monitoring service.)

R

realm

A logical collection of regions. Realms are isolated from each other and do not share any data. Your tenancy exists in a single realm and can access the regions that belong to that realm.

region

A collection of availability domains located in a single geographic location.

regional subnet

A subnet that spans all availability domains (ADs) in the region. Oracle recommends using regional subnets because they are more flexible and make it easier to implement failover across ADs. Compare with AD-specific subnets.

remote peering connection

A component on a dynamic routing gateway (DRG) for routing traffic to a remotely peered VCN. "Remote" peering means the two VCNs are in different regions. Compare with a local peering gateway.

remote VCN peering

The process of connecting two VCNs in different regions so that their resources can communicate without routing their traffic over the internet or through your on-premises network.

reserved public IP

A public IP address (and related properties) that you create in your tenancy and assign to your instances in a given region as you like. It persists in your tenancy until you delete it. It can be assigned to any private IP on a given VNIC, not just the primary private IP. Compare with ephemeral private IP.

resolution

The period between time windows, or the regularity at which time windows shift. (Monitoring service.) Example: 1 minute

resource

The cloud objects that your company's employees create and use when interacting with Oracle Cloud Infrastructure.

route table

Virtual route table for your VCN that provides mapping for the traffic from subnets via gateways to external destinations.

RPC

A component on a dynamic routing gateway (DRG) for routing traffic to a remotely peered VCN. "Remote" peering means the two VCNs are in different regions. Compare with a local peering gateway.

S

secondary IP address

An additional private IP you've added to a VNIC on an instance. Each VNIC automatically comes with a primary private IP that cannot be removed.

secondary VNIC

An additional VNIC you've added to an instance. Each instance automatically comes with a primary VNIC that cannot be removed.

security list

A list of virtual firewall rules for your VCN. Security lists consist of security rules that apply to traffic coming in and out of a VNIC. Security lists are configured at the subnet level, which means all VNICs in a given subnet are subject to the same security rules.

security rule

Virtual firewall rules used by security lists to control traffic at the packet level.

service gateway

An optional virtual router that you can add to your VCN. The gateway enables cloud resources to privately access Oracle services (such as Object Storage and Autonomous Database) without exposing the resources to the public internet.

shape

A template that determines the number of CPUs and the amount of memory allocated to a newly created instance.

statement

Policies can contain one or more individual statements. Each statement gives a group a certain type of access to certain resources in a particular compartment.

statistic

The aggregation function applied to the given set of raw data points. Example: SUM

subnet

Subdivision of your VCN used to separate your network into multiple smaller, distinct networks.

subscription

An endpoint for a topic. Published messages are sent to each subscription for a topic. Supported subscription protocols include Email and HTTPS (PagerDuty). (Notifications service.)

suppression

A configuration to avoid publishing messages during the specified time range. Useful for suspending alarm notifications during system maintenance. (Monitoring service.)

Swift password

(Deprecated. Use an auth token to authenticate with your Swift client.) Swift is the OpenStack object store service. A Swift password enables you to use an existing Swift client with Oracle Cloud Infrastructure Object Storage.

T

tenancy

The root compartment that contains all of your organization's compartments and other Oracle Cloud Infrastructure cloud resources.

tenant

The name assigned to a particular company's or organization's overall environment. Users provide their tenant when signing in to the Console.

topic

A communication channel for sending messages to the subscriptions in the topic. (Notifications service.)

transit routing

An advanced routing scenario that enables communication between an on-premises network and multiple virtual cloud networks (VCNs) over a single FastConnect or IPSec VPN.

trigger rule

The condition that must be met for the alarm to be in the firing state. A trigger rule can be based on a threshold or absence of a metric. (Monitoring service.)

U

user

An individual employee or system that needs to manage or use your company's Oracle Cloud Infrastructure resources.

V

VCN

A virtual version of a traditional network—including CIDRs, subnets, route tables, and gateways—on which your instance runs.

virtual circuit

Used with Oracle Cloud Infrastructure FastConnect. An isolated network path that runs over one or more physical network connections to provide a single, logical connection between the edge of your existing network and Oracle Cloud Infrastructure.

virtual cloud network

A virtual version of a traditional network—including CIDRs, subnets, route tables, and gateways—on which your instance runs.

virtual machine

A software-based emulation of a full computer that runs within a physical host computer.

virtual network interface card

A VNIC enables an instance to connect to a VCN and determines how the instance connects with endpoints inside and outside the VCN. Each instance automatically comes with a primary VNIC, and you can add secondary ones. Other types of cloud resources also automatically get a VNIC upon creation (examples: load balancers, DB systems).

VM

A software-based emulation of a full computer that runs within a physical host computer.

VNIC

A VNIC enables an instance to connect to a VCN and determines how the instance connects with endpoints inside and outside the VCN. Each instance automatically comes with a primary VNIC, and you can add secondary ones. Other types of cloud resources also automatically get a VNIC upon creation (examples: load balancers, DB systems).

volume

A detachable block storage device that allows you to dynamically expand the storage capacity of an instance.

W

work request

An object that reports on the current state of an asynchronous service request.

RELEASE NOTES

You can find the Oracle Cloud Infrastructure Release Notes online.